# Context-Aware Adaptive Hybrid Semantic Relatedness in Biomedical Science

by

Ehsan Emadzadeh

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2016 by the
Graduate Supervisory Committee:

Graciela Gonzalez, Chair
Robert Greenes
Matthew Scotch

ARIZONA STATE UNIVERSITY

May 2016

## ABSTRACT

Text mining of biomedical literature and clinical notes is a very active field of research in biomedical science. Semantic analysis is one of the core modules for different Natural Language Processing (NLP) solutions. Methods for calculating semantic relatedness of two concepts can be very useful in solutions solving different problems such as relationship extraction, ontology creation and question / answering [1–6]. Several techniques exist in calculating semantic relatedness of two concepts. These techniques utilize different knowledge sources and corpora. So far, researchers attempted to find the best hybrid method for each domain by combining semantic relatedness techniques and data sources manually. In this work, attempts were made to eliminate the needs for manually combining semantic relatedness methods targeting any new contexts or resources through proposing an automated method, which attempted to find the best combination of semantic relatedness techniques and resources to achieve the best semantic relatedness score in every context. This may help the research community find the best hybrid method for each context considering the available algorithms and resources.

# DEDICATION

To the love of my life, Azadeh, for supporting me all the way. To my parents for their encouragement and life-long dedication.

ACKNOWLEDGMENT

I wish to express my sincere gratitude to my supervisor Dr. Graciela Gonzalez whose expertise, constant support and guidance enabled me to find a topic that was a great interest to me.

I am grateful to all of my committee members, Dr. Robert Greenes and Dr. Matthew Scotch for their frequent feedback and guidance.

TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

Figure Page

## LIST OF EQUATIONS

# 1 INTRODUCTION

Biomedical text mining is an increasingly important area in biomedical studies. Massive amount of textual information in the forms of research papers and clinical notes are being generated every day. Medline alone indexed about 2000-4000 references each day since 2005[1]. With more than 5000 hospitals in the USA[2], we can estimate that a prodigious number of clinical notes are being generated daily. Despite the existing advances made so far in this field, there are still missing elements in biomedical text mining approaches leading to inefficient solutions for real-world needs [7].

To reach a point that a text mining tool can be truly effective in a clinical or research setting, computers should be able to understand deeper levels of semantics in text. As a result of deeper understanding of the meanings, further knowledge can be extracted from raw text providing a higher level of decision supports to experts. For example, if a system can extract concepts and relationship from a patient history correctly, it can extract a time-line of events expressed in the clinical notes or answer a physician's questions about a specific problem in a timely manner. The levels of abstraction in text analysis defines different tasks that we need to deal with in this field:

1. Lexical and Syntactical: it is related to structure of the text and its physical representation. Tasks like grammar parsing, tokenization, and sentence detection are in this category.

[1] http://www.nlm.nih.gov/pubs/factsheets/medline.html
[2] http://www.aha.org/research/rc/stat-studies/fast-facts.shtml

2. Semantic: it deals with the meaning of the text. Named entity recognition (NER), relationship extraction, normalization and reasoning are some of the tasks in this category.

3. Discourse: it considers a document as a whole and tries to find the perspective of author. Tasks like summarization fall into this category.

This work focuses on semantic analysis of biomedical text and proposes a new measure of semantic relatedness evaluated in two applications: 1. Normalizing adverse drug reactions in English colloquial text; and 2. Gene function extraction from publications. These two applications are not the only applications for semantic relatedness in natural language processing. Since the main focus of this work is a new semantic relatedness technique, the next section expands to semantic analysis. The proposed semantic relatedness method will be evaluated in two biomedical text mining applications; therefore, major challenges in biomedical text mining is discussed in the following section.

**Section 1.1 Semantic Analysis**

Semantic analysis is a sub-discipline of Natural Language Processing (NLP) which aims to make computers understand the meaning of a piece of natural language text. This can be very challenging since the definition of semantic is not completely clear. In the following sections I will provide a definition of semantic and explain how similarity and relatedness help us to quantify semantic.

**Definition of Semantics**

Semantics is the study of meaning. However, depending on the field it can have different interpretations. In linguistics, semantics is a branch of semiotic studies. Semiotic studies is the study of how the meaning are made and formally is defined as: “The science of

communication studied through the interpretation of signs and symbols as they operate in various fields, esp. language" [8] (p. 249). Linguistics branch semiotic studies into three categories:

1. Syntactics: it studies the structures of language. Syntactics is formally defined as: "The branch of semiotics that deals with the formal properties of signs and symbols" [9].
2. Semantics: it goes beyond structure and studies the relations between language signs and structure and what they refer to in real world. It is formally defined as: "The study of relationships between signs and symbols and what they represent. Also called semasiology" [9].
3. Pragmatics: it is very similar to semantics, but in addition to language it considers who is the speaker and who is the listener and the context the language signs are used. For instance when a window is open and someone says "It's very cold," it implies asking to close the window.

Based on the above definition for semantics, the meaning can be defined as the concepts that a phrase brings to mind. This is referred to as connotation: "The set of associations implied by a word in addition to its literal meaning" [9]. Related concepts come to mind all depend on the context where the phrase is used. For instance a word like bridge used in a construction fields brings completely different set of concepts than when it is used in a dentist office. In addition to connotation, sometime the phrase is referring exactly to a single concept (e.g. "Pacific Ocean"). This is also a part of semantic which is called denotation: "The most specific or direct meaning of a word, in contrast to its figurative or associated meanings" [9]. Connotation and denotation both define meaning. This work

focuses on semantic similarity and relatedness that focus more on connotation part of the meaning definition. As such, the next section discusses the differences between similarity and relatedness in more details and followed by a section which focuses more on existing computational techniques for calculating relatedness.

**Semantic Similarity and Relatedness**

As a new computational method for semantic relatedness, it is important to understand the existing computational techniques and difference between relatedness and similarity. Semantic relatedness involves quantifying connotations and is a generalized term to describe the semantic closeness of two word / phrase meanings. Moreover, semantic relatedness can be used to calculate nearly anything (*e.g.*, comparing two patients); but in this work, I will limit semantic relatedness to textual artifacts, which can be a set of terms or documents. A narrower term is "semantic similarity," which only includes synonymy and measure the degree of which two phrase representing the same concepts. Similarity does not include other lingual relationships such as antonymy and meronymy and therefore is subset of relatedness. For instance “winter” and “flu” are related but not similar.

Semantic relatedness techniques can be grouped into two main categories: topological relatedness and statistical relatedness. Topological methods uses the link between concepts and usually applies to external knowledge sources like ontologies. The statistical relatedness is based on the distributional hypothesis in linguistics [10,11] which indicates two concepts are similar if they appear in similar contexts. In many publications, statistical relatedness refers to “distributional semantic.” The next section discusses some computational techniques for calculating a numeric representation of semantic relatedness of two textual contents.

**Measure of Semantic Relatedness**

Understanding the relatedness of two concepts in a way that an expert understands their relatedness enables the computer to use (simulated) human judgments when dealing with text. Measure of Semantic Relatedness (MSR) functions tries to find relative meaning closeness between two concepts. These functions have wide usage in search engines [4,12], questions and answers [6,13], *etc*. The output of an MSR function is a value (usually normalized between 0 and 1) showing how much two given concepts are semantically related. Most of the MSR functions need a resource to calculate the relatedness. The resource's accuracy and completeness affect the performance of the MSR; therefore a suitable corpus should be found for every context. For example, if the MSR uses WordNet to calculate the similarity of "paroxysmal cough" and "nocturnal cough," the value would be near 0 because none of those concepts is in the WordNet set of defined words. Instead, if it uses MeSH instead of WordNet, the relatedness value would be very high since both appear in MeSH ontology. In addition, semantic relatedness depends highly on which context the judgment happens. For example, when we are doing a kidney disease study, "paroxysmal cough" and "nocturnal cough" are expected to be very similar, but in a pneumonia study, they are less similar [14].

MSR functions are based on statistical, vector or graph analysis. For example, Latent Semantic Analysis (LSA) [15] is a vector-based semantic similarity measure, Point-wise Mutual Information (PMI) [16] is a statistical semantic similarity measure, and Incremental Construction of an Associative Network (ICAN) [17] is a graph-based MSR method. Budiu *et al.* [18] compared LSA, PMI and Generalized LSA (GLSA) [19] using two corpora; the

TASA corpus[3] [20] and Stanford corpus[4] [21]. They performed two tests: a synonym test, and similarity rankings by humans.

For the synonym test, they used questions from three English exams: Test of English as a Foreign Language (TOEFL), English as a Second Language (ESL) and Readers Digest Word Power Vocabulary Test (RD). In a synonym test, they found that GLSA did better than others, and that, as the corpora size increased, the accuracy also increased. On the other hand, PMI did best on similarity ranking. Matveeva *et al.* [19] compared GLSA and PMI and showed that GLSA outperforms PMI for finding synonyms.

Table 1 compares strengths and weaknesses of LSA, PMI, ICAN and SRS. LSA is suitable for a situation where a specialized corpus is available (*e.g.*, discharge summaries) and we do not want any irrelevant contents in the calculation. PMI can be used for quick calculation of relatedness based on the distribution of data in a large dynamic corpus [22]. This makes PMI a better choice for use with large textual resources, which change frequently over time, like PubMed. As for ICAN, the model creation is extremely costly, but it is easy for humans to comprehend the model since it generates a graphical representation of information. It also provides asymmetric similarity measures that are useful for creating asymmetric relations (*e.g.* generalization relationship in ontology).

Some methods have tried to combine the individual functions to achieve higher performance. For instance, Semantic Relatedness Score (SRS) [23] combines MSRs to enhance syntactic matching algorithm. SRS is a hybrid system designed to improve

[3] "60,527 samples of text from 6,333 textbooks, works of literature, and popular works of fiction and nonfiction used in schools and colleges throughout the United States."

[4] Output of Stanford WebBase project, which tries to copy the whole web

existing MSRs for solving ontology matching problems. For example, integrating two epilepsy ontologies developed by two different experts can be an excellent application of SRS. Pederson et al.[24] compared six semantic similarity methods on 30 pairs of clinical terms and showed that the vector-based methods outperform others. Liu *et al.* [25] proposed a new hybrid system that combined vector methods and an existing ontology (compared UMLS and WordNet). He showed that the new system outperforms existing ones. Pivovarov *et al.* [14] combined vector-based, graph-based and dictionary-based methods for clinical concepts and showed the combination outperforms existing methods. Petrakis et al. [26], Rodriguez *et al.* [27] and Li *et al.* [28] tried to design MSR that can benefit from multiple resources and are not bound to a single resource. They all showed that the hybrid method outperforms single MSRs.

The existing hybrid systems were designed heuristically, and there is no claim that they can outperform other methods in every context. In addition, little work has been done to design a context-aware system that can be adjusted to a new context automatically. To the best of my knowledge, an Adaptive Semantic Analysis function that can adapt to a different context automatically is missing. The focus of this work is on a new adaptive semantic relatedness function that can benefit from new resources automatically, and find the best combination of resources and algorithms for each context by training a regression model. The proposed hybrid model will train a regression model using existing machine learning techniques and the trained model can be used as the semantic relatedness function. Table 1 lists strength and weakness of some MSR methods. The hybrid model proposed by this work tries to combine the benefits of all different technique and reduce the weaknesses at the same time. For instance, the proposed method can benefit from vector-based, statistical

and graph based methods at the same time. This means it has the strength of all individual technique (e.g. LSA, PMI, and ICAN) and remove some weaknesses such as support multi-word and n-order similarity. The main downside of the proposed method is that it requires more processing power than each individual method. The proposed system will be evaluated from two aspects:

1. Correlation of the relatedness scores and experts' opinion.
2. ADR normalization improvement caused by the proposed relatedness function (As an example of downstream application).

**Table 1. Comparing pros and cons of some MSR techniques.**

| **MSR** | **Strength** | **Weakness** |
|---|---|---|
| LSA | - Vector-based<br>- Can compare multi-word (phrases and documents)<br>- N-order similarity | - Non-incremental vocabulary set, limited to the corpus<br>- Pre-processing time is too long |
| PMI | - Large vocabulary set support<br>- can use any search engine (like Google)<br>- no pre-processing<br>- Specialize easily (by limiting the search engine to use certain websites) | - Hard to support multi-world comparison<br>- Search engine speed affects the performance<br>- Costly (search engines charge money for each search query)<br>- Only first order similarity and does not support n-order similarity |
| ICAN | - Incremental vocabulary set<br>- Network-based measure<br>- Spreading activation, which is | - Costly pre-processing<br>- Hard to support the multi-world similarity |

| | | |
|---|---|---|
| | useful for n-order relatedness<br>- Asymmetric similarity | |
| SRS | - Hybrid<br>- Support multi-world<br>- Optimized for ontology mapping | - Domain-specific and not generalizable<br>- Costly calculation |

## Section 1.2 Current Challenges in Biomedical Text Mining

### Clinical Notes Mining

Clinical notes are the textual information about patients, created by health-care professionals, to record patients' clinical status during the course of a hospitalization or outpatient care. Clinical notes can include a variety of reports such as radiology reports, progress reports, and admission notes. Each sub-language in medicine has unique characteristics and can bring a lot of new challenges for the existing solutions [29]. Information extraction systems can help access information in textual notes easier to reduce experts' workload. In addition, automatically extracted information from clinical notes can prevent possible human errors and foster research. If the output of the system is supposed to be used for making a clinical decision, then the information extraction output should have extremely high accuracy and recall, otherwise the expert will not trust the system.

### Scientific Literature Mining

Research papers, specifically biomedical research papers, are the textual reports that detail research methodology and results. Considering the large number of publications each year,

an effective information extraction system can be very useful for other researches / clinicians to study outcomes of previous works. For example, finding all related works that address about a gene expression can help a new researcher to focus on highly related works. Even though high precision and recall are desirable in this information extraction task, the tolerance of error is much higher than with clinical notes. Since a user will review each returned piece of information from research papers, having a good recall rate is favored over high precision and low recall. Some of the critical active problems in literature mining are: concept extraction [30–32], relationship extraction [33–35], fact extraction [13,36] and summarization [37]. In the following subsection of gene function extraction, an example of existing problems in literature mining is discussed. In addition, in the following chapters, a semantic relatedness based solution will be proposed for gene function extraction.

***Gene Function Extraction***

Biomedical literature mining aims to reduce manual labor and provide enriched information that can empower advances in medical research and treatments. Lu et al. [38] demonstrated that there is an increasing interest to use text mining techniques for curation workflows. Currently, literature curation is challenged by lack of automated annotation techniques, particularly in Gene Ontology annotations [38]. In medical informatics alone, the number of indexed articles has increased by an average of 12% each year between 1987 and 2006 [39][40] with close to 20 million articles indexed in PubMed in 2013. With an increasing number of publications detailing complex information, the need to have reliable and generalizable computational techniques increases rapidly.

Finding gene functions discussed in literature is crucial to genomic information extraction. Currently, tagging the gene functions in published literature is mainly a

manual process. Curators find gene function evidence by reviewing each sentence in relevant articles and mapping the results to standard ontologies, and, specifically for this problem, to the Gene Ontology (GO) [41] as a controlled vocabulary of gene functions.

The BioCreative IV GO workshop [43] aims to automate gene function curation though computational methods. With a focus on gene functions, it includes two sub tasks: a) Retrieving GO evidence sentences for relevant genes; and b) Predicting GO terms for relevant genes. Sub task of the goal in b) is to find the related gene functions (GO terms) in a set of genes discussed in an article. More details about the shared task and the corpus can be found in Auken et al. [43]. This task is very similar to BioCreative I subtask 2.2 held in 2004 [44] in which Blaschke et al. [44] summarized the results for BioCreative I. For subtask 2.2, the highest precision was reported to be 34.62% [45]. BioCreative IV GO subtask 2 includes an annotated corpus to enable the measurement of recall and F-measures. Couto et al. [46] used an information retrieval technique to find related sentences and GO terms. Furthermore, Chiang et al. [45] combined sentence classification with pattern mining, and Ray et al. [47] proposed a solution based on probabilistic model and Naïve Bayes classifier.

Most of the participants in the previous related task focused on information content and statistical models combined with machine learning. Here, an unsupervised method based on distributional semantic similarity is proposed and compares an existing measure of semantic relatedness with the proposed hybrid semantic relatedness method. More details about the method and results can be found in Chapter 2 and 5.

## Social Media Text Mining

Social media are network based software tools that allow people to share and discuss any information inside a virtual community. Kaplan *et al.* [48] (p. 61) defines social media as "a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user-generated content". The main characteristics that differentiate social media from other types of media are the interactive user-generated content and the use of Web or mobile networks. Social media has significantly changed the way businesses, communities and individuals communicate. For example, businesses can get feedback directly from their end-user on a product without performing costly surveys. For individuals, on the other hand, social media can be very helpful in different ways. For example, people with the same health problem can help each other by forming specialized groups and sharing their experience (e.g. www.patientslikeme.com). Social media is the main focus of Technoself [49] studies which focus on studying human identity in a technology world.

Different social media have been invented for various applications. The following section reviews the classification of social media and how they helped or can potentially help advance medical knowledge.

### *Social Media Classification and Potential Health Application*

There are various types of social media technologies including blogs, forums, microblogs, photo sharing, products / services reviews, social gaming, social networks, video sharing and virtual worlds [50]. In the following sections, some important types of social media are reviewed with their potential applications in medical domains.

*Blogs*

A blog or weblog is a site on the World Wide Web [51] which consists of multiple posts. Each post is mainly textual but may contain other types of media such as pictures and videos. A blog is mainly about a single subject and, in its early ages, only a single person managed it. Multi-Author Blogs (MABs) were introduced years after the blog invention. Blogs are still a popular method of sharing information but other types of social media such as microblogging which has become more popular. Health care professional uses blogs to share recent publications, discuss professional matters or increase public health awareness (e.g., http://drwes.blogspot.com.au/, and http://casesblog.blogspot.com/).

*Microblogs*

Microblog is a specific type of blog, which consists of more frequently shorter posts. It also encourages interaction between users by allowing them to mention other users in their posts and reply to other persons' posts. By having a short length and easy accessibility, Microblogs allow users to share their thoughts, feeling and information at a moment. There are many microblogging services such as Twitter, Tumblr, Plurk, etc. Other social media websites, such as Facebook, LinkedIn, and Google+, provide microblogging as well via status update. Data from microblogs have been used in the past for tasks such as studying smoking cessation patterns on Facebook [52], identifying user social circles with common medical experiences (like drug abuse) [53], and monitoring malpractice in Twitter [54]. In addition, recent research has utilized social media for the monitoring of ADRs from prescribed medications [55–58].

*Forums*

Forums are online bulletin, which consist of several conversations (threads) in different categories. Each thread is about a single topic or question and users can post to the thread about the subject of the thread. Forums are managed by a forum moderator to keep the forum organized (*e.g.,* make sure every post is relevant to the thread's subject). Forums have become popular for question / answering for online communities. Thousands of medical related forums exist and they involve both professionals and patients. Some forums are about specific health conditions and patients share their experience or ask related questions. Forums are excellent information resources for various purposes and many text mining techniques have been proposed to extract knowledge from forums: finding hot-spot topics [59], thread genre classification [60] and search on forum [61], just to name a few.

*Photo Sharing*

Even though most of social networking services support photo sharing, but some social networks are solely centered on photo sharing (*e.g.,* flickr, Instagram, Google Picasa). In these services a user can share photos and other users can comment about the photo. However not much has been done to mine photo sharing service, but potentially they can be helpful in finding outbreaks or public health metrics.

*Video Sharing*

Video sharing services are social media focused on videos, which are usually small clips. Websites, like YouTube and Vimeo, provide platforms for users to upload videos recorded by their mobile or professional cameras, and other members can comment about a video. Videos and comments in the video sharing services can be leveraged to find out what users

are interested or concerned about, which can be used alongside other public health metrics to make proper decision to improve public health.

*Products / Services Review*

Some social media center on reviews of products and services. In these services the topic of discussion is a product or quality of services of a company. For example, drugs.com and amazon.com are two services that users can post reviews about products. This can be very useful in the understanding of pros and cons of each product. Specifically in medical domain, this can be extremely helpful to find out if specific drug or treatment was effective and what were the adverse side effects. The advantages of these services over other type of social media are that the comments are about specific product (*e.g.* a drug); and it makes it easier to understand the subject of comments.

***Social Media and Pharmacovigilance***

Pharmacovigilance is defined as "the science and activities relating to the detection, assessment, understanding and prevention of adverse effects or any other possible drug-related problems" [62] (p. 828). The primary focus of pharmacovigilance is the monitoring of adverse drug reactions (ADRs). Due to the various limitations of pre-approval clinical trials, it is not possible to assess the consequences of the use of a particular drug before it is released [63]. As such, ADRs caused by prescription drugs is currently considered to be a major public health problem and various ADR monitoring mechanisms are currently in place, such as voluntary reporting systems, electronic health records, and, relatively recently, social media [64].

Social media has emerged as an important source of information for various public health monitoring tasks. The increasing interest in social media is largely because of the vast

abundance of data that it contains—data that is directly generated by consumers. Data from social networks have been used in the past for tasks such as studying smoking cessation patterns on Facebook [52], identifying user social circles with common medical experiences (like drug abuse) [53], and monitoring malpractice [54]. In addition, recent research has utilized social media for the monitoring of ADRs from prescribed medications [64]. From the perspective of pharmacovigilance, social media acts as a platform of paramount importance, since it has been shown in the past research that users discuss their health-related experiences, including the use of prescription drugs, side effects and treatments on a regular basis. Users tend to share their views with others facing similar problems / results, which make such social networks unique and robust sources of information about health, drugs and treatments [64].

In Summary, we reviewed semantic analysis from linguist perspective and how it is measure with computational methods. Multiple existing computational techniques to calculate semantic relatedness were compared. In addition three main category of biomedical text mining problem were discussed. In next chapter we introduce a new hybrid semantic relatedness method. Also two use cases where the proposed hybrid model is used in the solution will be discussed.

# 2 METHODS

This chapter introduces a new method of semantic relatedness which attempts to minimize errors of each individual MSR by combining them using a machine learning regression model. In addition two case studies for the proposed method are discussed: adverse drug reaction normalization and gene function extraction. These two case studies are used to evaluate the new semantic relatedness method. The case studies are selected to evaluate the new technique in two important types of biomedical texts: scientific biological literature and colloquial texts. To show the generalizability of the proposed method in various contexts, the selected problems are completely different types (one is normalization and one is concept extraction). Semantic relatedness techniques can be used as part of solutions for various problems, and here we show the application in two use cases to demonstrate the effectiveness and generalizability of the proposed technique. In addition the selected use cases are from our previous works which makes the comparison easier and more reliable.

## Section 2.1 Adaptive Semantic Analysis (ASA)

Some of the popular MSRs are compared in the first chapter. Different MSRs have been applied to various underlying models to solve the semantic relatedness problem. These models bring advantages and disadvantages to their solutions. Existing models can be categorized as:

- Vector-based: The methods in this category represent each phrase or word as a real number vector (or one dimension matrix). The relatedness of two phrases

or words is the distance between their vectors. The distance can be calculated using various methods such as cosine distance and Euclidean distance.

- Statistical: The methods only based on numbers of co-occurrences are in the statistical category.
- Graph based: The MSRs in this category utilize a graph data structure to calculate the relatedness of two concepts. Ontologies and linked web pages are examples of usable graph data structure for semantic relatedness.
- Hybrid models: Hybrid models combine methods from any of the above categories to leverage various knowledge formats and optimize the outcome.

Each category is discussed in Emadzadeh et al. [22] in more details. The hybrid category is the main focus here since the proposed system is a new adaptive hybrid MSR. There are several methods that use combinations of each individual MSR method (refers to as MSR kernels) and the goal is to reduce the disadvantages of each model by combining them. Different techniques have been proposed to combine MSR kernels (as we discussed in the introduction section). The new proposed hybrid system can accept any number of MSRs in various categories as input. The support for accepting heterogeneous MSRs allows the proposed hybrid method to leverage heterogeneous information sources in a single system. To list a few of them, the following are some of the MSRs which can be used in the proposed hybrid method:

- LSA [15]
- GLSA [19]
- PMI [16]

- Edge counting [65]
- Normalized path-length [66]
- Wu's scaled measure [67]
- Gloss vector [68]
- Gloss overlap [69]
- Extended gloss overlap [70]
- Normalized Google Distance (NGD) [71]

Most of MSR implementation are borrowed from Semantic Measures Library[72].Adaptive Semantic Analysis (ASA) is a new adaptable MSR technique for calculating semantic similarity of two concepts. ASA needs to be trained on a limited number of concept pairs for any new context, and it returns a normalized value between 0 and 1 that shows how strong the two concepts in each pair are semantically related (1 shows a strong relation, and 0 no relation). One of the challenges of calculating semantic relatedness is the need to consider the context for calculating the result. In ASA, the context is defined by two contextual variables:

1. Text type (e.g. "clinical notes");
2. Entities type (e.g. "problem-problem").

For instance, we want to find semantic relatedness for a new context where text type is "radiology report" and entities type is "test-problem". ASA gets the context ("radiology report", and "test-problem") and some training examples of semantic relatedness in that context as input; and finds the optimal combination of available semantic relatedness techniques. This technique allows us to use new corpora and knowledge sources without

requiring large amounts of annotation, which is particularly useful when dealing with a new problem. ASA combines statistical and graph-based techniques; as a result, it can benefit from both structured knowledge bases and textual corpora. ASA is not a domain-specific method and can adapt to various domains using the existing domain-specific knowledge-bases and corpora. Table 2 shows examples of MSR and resource matches. The matching of resources and MSR is done automatically by considering the requirements of each MSR kernel. This allows the addition of any new resources to the system easily and includes any new MSRs by defining their resource specifications.

**Table 2. An example of MSRs and Resources that is matched initially in the system. The table is dynamic, and the matching is done automatically by the system.**

| | **LSA** [15], GLSA [19], **PMI** [16] | Edge counting [65], Normalized path-length [66], Wu scaled measure [67] | Gloss vector [68], gloss overlap [69], extended gloss overlap [70] | Normalized Google Distance (NGD) [71] |
|---|---|---|---|---|
| **GENIA corpus** | √ | | | |
| **i2b2 2007 clinical notes corpus** | √ | | | |
| WordNet | | √ | √ | |
| UMLS | | √ | | |
| OBO | | √ | | |
| MeSH | | √ | √ | |
| Wikipedia | | | √ | |
| **Google Search API** | | | | √ |
| **Yahoo Search API** | | | | √ |
| Pubmed Search API | | | | √ |

$$AHMSR(context, C_1, C_2) = f(C_1, C_2, context, MSRs, DataSources)$$

Where:

$$context = (texttype, entitistype)$$

The method is designed to easily adapt to new knowledge sources or corpora and adjust parameters accordingly. Also, it can be trained for a new text type or entity type.

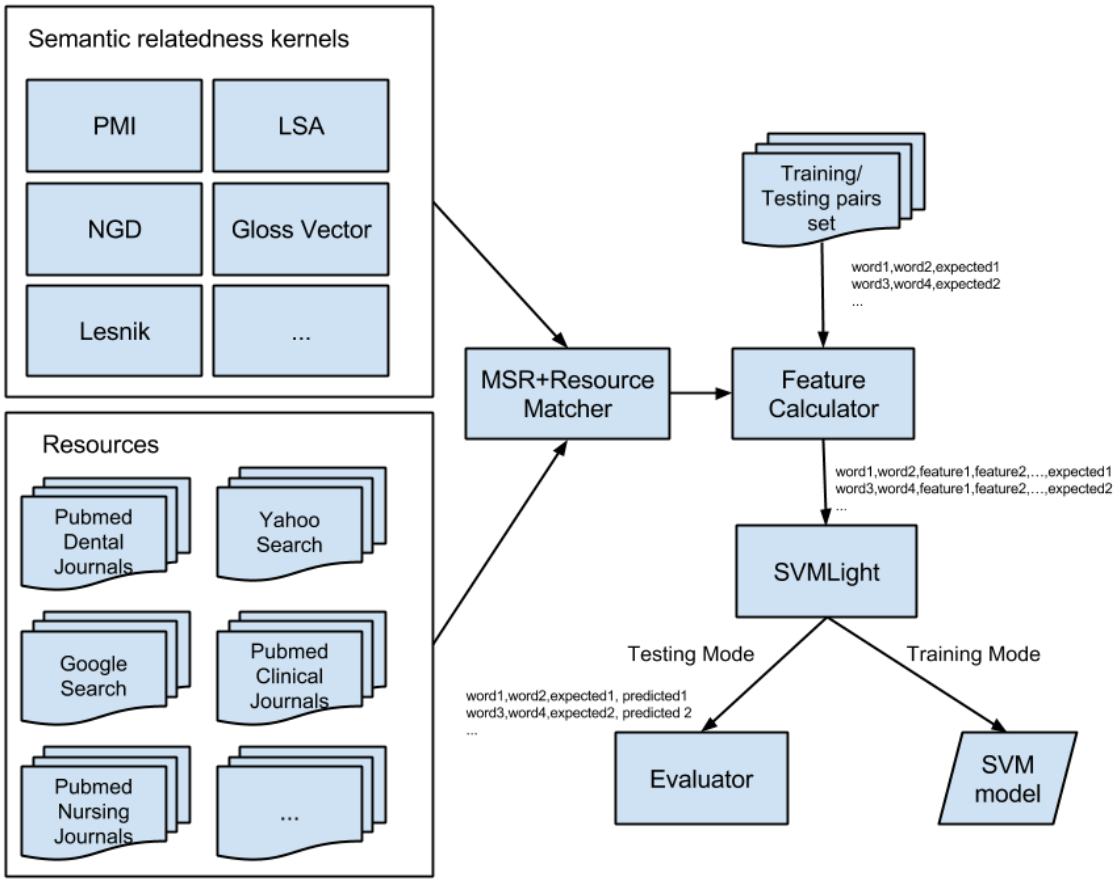


**Figure 1. Overall architecture of the ASA technique. For each pair, the training set "Feature Calculator" calculates features. Features are returned value from each MSR combined with different corpus. For example, one feature can be semantic relatedness returned for a pair by LSA-I2B2ClinicalNotes. After feature calculation, the regression model (SVM) will be trained, and the model will be evaluated against the test set.**

**Creating the Regression Model**

ASA uses machine learning to find the optimal combination of semantic relatedness functions for each context, based on the set of calculated features. For each pair in a training or testing set, these features are calculated. The features are MSRs values calculated based on the available resources (listed in Table 2). Each MSR can return different values when applied to different resources. Therefore, we can have many features being generated for each MSR combined with different resources. For example, we can have several feature using PMI and different resources, like PMI-GENIA and PMI-I2B2ClinicalNotes. Feature values are the semantic relatedness that are returned by each method executed on the given resources. Since MSR and information resources are dynamic and can be added or removed from the system, the feature set becomes dynamic and can vary in the future. The expected value from regression function is the semantic relatedness of two concepts in the given context. SVM ($SVM^{Light}$ [73]) was used to create the regression model but other models such as neural network can be used and explored. Each of the MSR kernels mentioned above are explained briefly in the following sections.

**Adding a New Information Resource or MSR**

Each informational resource is described in the system with a configuration entry which specifies the type of resources. A resource can be a graph, a text corpus, a dictionary or a search API. Similarly, MSRs are defined in the configuration with their required resource types noted. For example, the Edge count method requires a graph-based resource. This allows the matcher module to find suitable resources for each MSR dynamically. Adding a new resource or MSR is as easy as adding a new configuration entry to the system and

training the system again. This means new resources or algorithms can be injected to ASA without changing the method.

**Latent Semantic Analysis (LSA)**

LSA [74] uses a term-document frequency matrix to estimate semantic similarity of two texts. The typical technique for creating the vector of term weights is to use TF-IDF weighting. The TF-IDF equation (

Equation **1** has two parts: Term Frequency (TF) and Inverse Document Frequency (IDF). The TF part is shown in Equation 2 and the IDF in Equation 3.

$$tfidf_{i,j} = tf_{i,j} * idf_i$$

Equation 1. Term frequency-inverse document frequency

$$tf_{i,j} = {n_{i,j}}\Big/{\sum_{k=1}^{termsInDoc_j} n_{k,j}}$$

Equation 2. Term Frequency

$$idf_i = {|D|}\Big/{|\{d: t_i \in d\}|}$$

Equation 3. Inverse document frequency

Where $n_{i,j}$ is occurrences of $term_i$ in$document_j$, $|D|$ is the total number of documents in the corpus, and $|\{d{:}t_i{\in}d\}|$ is the number of documents where the term $t_i$ appears.

However, TF-IDF is the most popular technique for creating a term-document matrix, but other methods also can be used for creating such matrix. For example, the term-document matrix can be made simply from the number of occurrences of terms in documents. Then LSA harvests the matrix using Singular Value Decomposition (SVD) [75], by selecting the k best SVD values. Then using SVD, it splits the term-document matrix into two matrices, one for terms and one for documents as follows:

$$X = T_k * S_k * P_k^t$$

Equation 4. Singular Value Decomposition (SVD) used to convert term-document matrix to term vectors matrix

Where *X* is the term-document matrix and $T_k$ rows are the term vectors in LSA, space and columns in $P_k$ are document vectors. Using the term matrix$T_k$, the similarity of two terms are the cosine value of the relevant rows.

**Generalized LSA (GLSA)**

Generalized LSA [19] extends LSA by focusing on term vectors, in contrast with LSA which uses a dual document-term representation. It reduces the dimensionality with SVD similar to LSA, but instead it uses the term-term matrix (see Equation 4). LSA can be seen as a special case of GLSA. By these changes, GLSA removed the effects of documents on meaning similarity of words; and instead uses PMI for calculating similarity between terms. It makes a term-term matrix using PMI similarity values. GLSA combines vector and statistical models.

**Wordnet Gloss Vector**

Several methods have been proposed based on WordNet. WordNet Gloss Vector generates a vector for each term in WordNet using its gloss definition (the brief explanation of a word in WordNet). Then it uses those vectors to calculate similarity between terms. Another method is based on the graph distance of two nodes in a WordNet IS-A graph [66]. There are other algorithms based on WordNet as well [76–78].

**Pointwise Mutual Information (PMI)**

Pointwise Mutual Information [16] or specific mutual information is a statistically-based method. Equation 5 shows how PMI calculates similarity between two inputs.

$$PMI(x,y) = \log {P(x,y)}\Big/{P(x)P(y)}$$

Equation 5. Pointwise Mutual Information

The inputs, x and y, are two given terms. P(x, y) is the probability of co-occurrence for the joint distribution, and p(x) and p(y) show the appearance probability of x and y individually. To calculate PMI, we need to know p(x, y) which is calculated based on a given corpus. An example of p(x, y) is displayed in Table 3 (x="gene" and y="expression", using PubMed as the corpus). The second column shows the total number of returned documents in the search results. Having p(x, y), p(x) and p(y) calculating the PMI similarity value is straightforward. Using numbers in Table 3:

$$PMI(\text{“}gene\text{", “expression"}) = \log \frac{0.044}{0.085*0.07} = 7.39$$

Equation 6. Example of Pointwise Mutual Information

Equation 6 calculates PMI of two terms x="gene" and y="expression" using PubMed as the corpus. Here, p(x, y) is term occurrence probability joint distribution, and p(x) and p(y) show the probability distributions of x and y individually. Assuming total documents in the corpus is 22 million.

**Table 3. An example for PMI calculation**. *P(x, y)* shows in what percentage of documents *x* and *y* appeared together. To calculate this probability, we need to know how many documents exist in the corpus and in how many of them both keywords exist. This enables the method to use any search engine with a corpus. Because we know the total number of documents indexed by a search engine[5], and the total number of search results for "x AND y" queries, we can calculate p(x, y) without any further processing of documents. This makes PMI faster compared to LSA and makes it scalable to use with very large corpora. Note that the value of PMI is not normalized between 0 and 1. The performance of PMI, like other corpus-based methods, is dependent on corpus richness and quality.

| **Query** | **Returned Documents** | **P(x, y)** |
|---|---|---|
| Gene | 1695952 | 0.085 |
| Expression | 1409462 | 0.07 |
| (gene) AND expression | 871240 | 0.044 |

**Normalized Google Distance (NGD)**

This method is based on a number of results returned by Google for given terms. This is a statistical method and depends on the Google distance formula (Equation 7).

$$NGD(x,y) = \frac{max\{log\ f(x), log\ f(y)\} - log\ f(x,y)}{\log M - \min\{log\ f(x), log\ f(y)\}}$$

Equation 7. Normalized Google Distance

[5] Check this website to see famous search engine indexed document statistics http://www.worldwidewebsize.com/

Where $M$ is the total number of web-pages indexed by Google. Here, $f(x)$ and $f(y)$ are the number of hits for search terms $x$ and $y$; and $f(x, y)$ is the number of web pages on which both $x$ and $y$ occur together. Any other search engines can be used instead of Google, and the result of this method is also called Normalized Search Similarity (NSS).

**Section 2.2 Case Study 1: Adverse Drug Reaction Normalization**

For downstream evaluation of the proposed sematic relatedness function, it can be used inside a novel ADR normalization system. The last section describes how social media is used for pharmacovigilance. However, while significant progress has been made in ADR text classification [79] and ADR mention extraction [80], the normalizing of user posted ADR mentions into a predefined set of concepts is still an unaddressed problem.

For the extrinsic evaluation of ASA, the task of normalizing distinct ADR mentions are addressed to more generalized concepts. This is a crucial task, which needs to be performed following the task of automatic ADR extraction [80]. For social media data, this is particularly important because users tend to express their problems often using non-medical terms. For example, considering the following user posts:

- "<DRUG NAME> makes me having the sleeping schedule of a vampire."
- "<DRUG NAME> evidently doesn't care about my bed time"
- "...wired! Not sleeping tonight. #<DRUG NAME>"

It can be seen from the above example that the same ADR is expressed in multiple ways. In our corpus, the mappings of the expressions to concepts in Unified Medical Language

System (UMLS) [81] ontology are performed manually by domain experts. The target of this task is to automatically identify the most appropriate concept for a distinct ADR mentions. I propose a natural language processing pipeline for normalizing the extracted ADR in tweets to UMLS concepts.

**Concept Normalization**

The task of normalization of ADRs involves assigning unique identifiers to distinct ADR mentions with different lexical variants of the same concept having the same ID. The IDs are derived from any lexicon or knowledge base with sufficient coverage. In the case of our research, we use the UMLS concept identifiers (henceforth: CUIs) to uniquely specify each ADR concept. The UMLS provides a vast vocabulary of medical concepts and the broad semantic groups into which the concepts can be classified. Each UMLS concept is assigned a unique ID, which represents all the lexical variants of the concept. For example, all synonyms of the concept hypertension (e.g., hypertensive disorder, high blood pressure, high bp and so on) are assigned the ID C0020538. The UMLS Metathesaurus, due to its comprehensive coverage of medical terminologies, has been used to build corpora specialized for normalization in the past [82–84].

The task of medical concept normalization can be regarded as a sub-field of biomedical named entity recognition (NER). Due to the abundance of text based medical data available, NER and concept normalization have seen significant research in the medical domain primarily through challenges such as BioCreative [85], BioNLP [86], TREC [87], and i2b2 [88]. Built on from these initiatives, the problem of concept normalization has seen substantial work for genes and proteins. Majority of the research on concept normalization relies on some variants of dictionary lookup techniques and various string

matching algorithms. Machine learning techniques have recently been employed, but mostly in the form of filtering techniques to choose the right candidates for normalization [89]. A number of approaches [90] rely on the use of tools / lexicons such as MetaMap [91] as a first step for the detection of concepts. Due to the advances in machine learning techniques and also the increasing availability of annotated data, recent approaches tend to apply learning based algorithms to improve mere dictionary lookup techniques.

Very recently, Leaman et al. [89] applied pairwise learning from a specialized disease corpora for disease name normalization. Prior works have involved list-wise learning, which learns the best list of objects associated with a concept and returns the list rather than a single object, for tasks such as gene name normalization [92,93], graph-based normalization [94], conditional random fields [95], regression based methods [96], and semantic similarity based techniques [97]. Semantic similarity or relatedness is a measure that shows how similar two concepts are. Such measures are often used for word sense disambiguation [70] where the term and its context information are utilized to assign a meaning to it. A number of techniques for computing semantic relatedness among medical entities have been proposed and compared in the past [98], some of which are mentioned in the next section. However, to the best of our knowledge, measures of semantic relatedness have not been previously used for normalizing ADR mentions.

**Social Media Text Normalization**

While the task of normalization of medical concepts is itself quite challenging, in our case, the problem is exacerbated by the fact that our data originate from social media. Social media data are notoriously noisy [99]. And while this hampers the performance of natural language processing (NLP) techniques, it is also the primary motivation behind the

implementation of techniques for automatic correction and normalization of medical concepts in this type of text. Typos, ad hoc abbreviations, phonetic substitutions, use of colloquial language, ungrammatical structures and even the use of emoticons make social media text significantly different from texts from other sources [99]. Past work on normalization of social media text focused at the lexical level, and has similarities to spell checking techniques with the primary difference that out of vocabulary terms in social media text are often intentionally generated. Text messages have been used as input data for normalization models, and various error models have been proposed, such as Hidden Markov Models [100] and noisy channel models [101]. Similar approaches targeted purely towards lexical normalization have been attempted on social media texts as well [102,103].

**ADR Normalization with ASA**

The goal of this normalization task is to find the UMLS concept ID related to a text segment in a tweet which is tagged as an ADR. For example, in the tweet: "*had 2 quit job: tendons in lots of pain,*" the phrase "*tendons in lots of pain*" is tagged as an ADR. The goal of our system is to normalize the annotated text to a concept in UMLS, which in this example is "c0231529-tenalgia." The following diagram shows the overall pipeline of the proposed normalization system. The system consists of Syntactic and Semantic matchers, Synonym Normalization and Evaluation components. As soon as a matcher finds a match, the remaining matchers in the pipeline will be skipped and the flow goes to Synonym Normalization and Evaluation components.

For evaluation, we use a previously annotated corpus of 2008 tweets about drugs. The corpus was generated by using Twitter API to search for tweets that contain the name of selected drugs. The dataset includes 1544 annotations using 345 unique concepts, of which

1272 are ADRs, 239 are "Indication" and 32 are "Drug." In this work we did not differentiate between annotation types (e.g., ADR or Indication) and attempted to normalize all types using the same pipeline. More information about the corpus and annotation can be found in Ginn et al. [58].

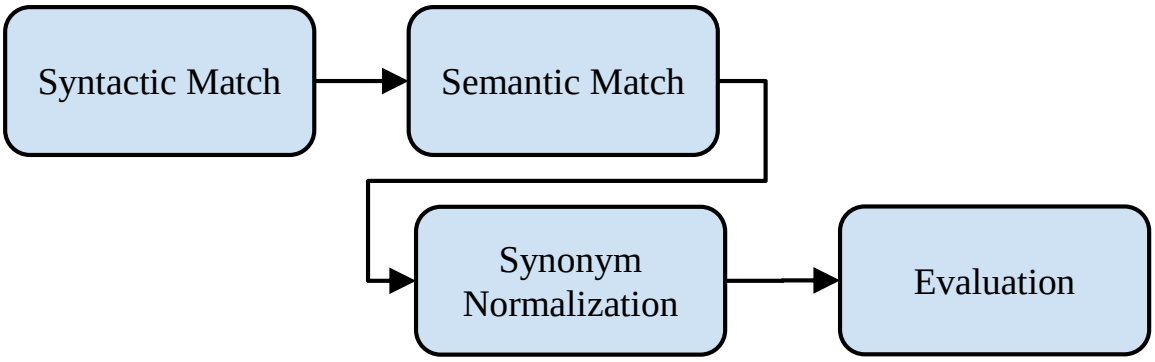


**Figure 2. Proposed normalization system's pipeline. As soon as a matcher finds a match, the flow skips to Synonym Normalization and Evaluation components.**

***Syntactic Match***

The first step in our normalization pipeline involves syntactic or lexical matching with concept names in UMLS. This part of the pipeline involves two steps: exact match, and definition match. An exact match happens when an ADR mention in a user post exactly matches a UMLS concept name. This simple match can detect many easy matches when standard terminologies are used by the users. However, in many cases in informal text, ADR mentions are misspelt, and exact matches are not possible. Some of these misspellings can be caught by a simple pre-processing. Unnecessary character repetition can be removed; for instance, in the tweet "*I feel siiiiiiiiiiiiiick,*" "siiiiiiiiiiiiiick" is matched with UMLS concept "c0231218-sick."

The next step in syntactic matching utilizes the formal definitions of UMLS concepts. The UMLS metathesaurus provides one or more definitions for each concept. The definition is a passage that describes the concept in plain English. We used this information, in the semantic similarity component later in the pipeline, to create semantic vectors and calculate the similarity values. In the syntactic matching module, we checked if the mention appears in the definition of a single concept in UMLS. If it only appears in the definition of one concept, the mention is normalized to the concept. In most of the cases a phrase appears in the definition of many concepts and no conclusion can be made.

***Semantic Match***

ADR concepts that are not normalized by the syntactic matching components are passed on for semantic matching. The primary task of this component is to compute the similarities of potential ADR concepts with the UMLS concepts. We experiment with several measure of semantic relatedness (MSR) methods. In this module, an MSR method is used to find semantic relatedness of a mention and a subset of concepts in UMLS. The most similar concept, with a similarity above a specified threshold, will be chosen as the concept of the mention. We evaluated the following MSR methods: Latent Semantic Analysis (LSA) [15], and our proposed hybrid method ASA. In semantic matchers, only the UMLS concepts which are used in the annotation are considered for the prediction.

*LSA [15]*

It uses a term-document frequency matrix to estimate semantic similarity of two segments of text. LSA then harvests the matrix using Singular Value Decomposition (SVD) by selecting the k best SVD values. More details about LSA technique and various weighting

techniques can be found in [15,104–106]. In our system, for the first step, the term vector space is generated from a corpus of plain text documents. Then this vector space is used to find a representative vector for each UMLS concept. The UMLS concept names are used to search for term vectors in the vector space. We evaluated some of the corpora which are listed in Table 4 for creating UMLS concepts representative vectors.

After finding a representative vector for each UMLS concept, we searched for a representative vector for each annotated text in the same vector space. The cosine similarity of each concept's vector and the annotated text's vector was calculated. The concept with the highest similarity to the ADR was chosen as the normalized concept if the cosine similarity was above a certain threshold (=>0.8).

*ASA*

It is our proposed hybrid MSR framework for calculating semantic similarity of concepts. The inputs of the framework are a list of resources (e.g., a corpus of PubMed abstracts or the UMLS ontology) and MSR kernels. Since manually finding the best resource for creating MSR models for a specific problem can be challenging, ASA trains a regression model to find the best combination of resources and MSR methods for a certain problem. For training the regression model, we prepared the training set from a subset of annotation (50% of the annotation). For each annotated text we created training examples for the annotated text and the UMLS concept names of the assigned concept with expected similarity of 100. For each annotation we generated 10 negative examples from the annotated text to random concepts in UMLS with expected similarity of 0. Figure 3 shows an example how ASA training examples are generated.

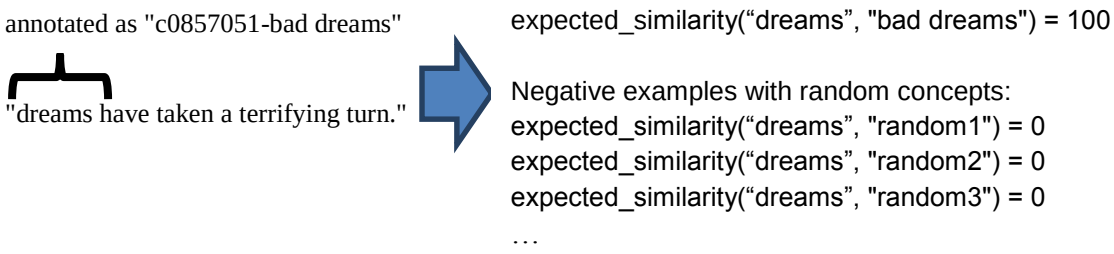


**Figure 3. This Shows How ASA Training Examples Are Created from ADR Normalization Annotations.**

The ratio of negative to positive examples can affect how the ASA regression model is trained. We used SVM with a linear kernel as the regression model, and trained ASA with the resources listed in Table 4 and LSA as the only MSR. We refrained from adding additional MSR methods as the intent of this experiment to study the effect of using the regression model with a single MSR and various additional resources. These resources are described in the next section.

After ASA was trained, the regression model was used to calculate the similarity of an annotated text to UMLS concept names. First, testing instances between the phrase and a set of selected UMLS concepts names were created. To limit the search space, we only used concepts with annotation frequencies of 3 or more for creating the test instances for ASA. Following that, for each example, the features were calculated. The features are all possible MSR and resource combinations defined in the system setup (e.g., LSA with Pubmed). Next the regression model was run on the test instances to calculate the similarity of the annotated text and each UMLS concept. The concept with the highest similarity and above a certain threshold ([>=90], note that the maximum and minimum similarities in the training set are 0 and 100), was chosen as the normalized concept. Since the method has to

calculate several semantic similarities for normalizing each annotated text, the process is slower than using a single MSR.

*CORPORA*

The two semantic matching techniques discussed above require data from suitable corpora to generate their models. We used three textual corpora generated from three different queries on PubMed (provided as special queries: http://www.nlm.nih.gov/bsd/special_queries.html): Dental Journals (PubMed query: "(jsubsetd[text])"), Nursing Journals (PubMed query: "(jsubsetn[text])") and Systematic Reviews (The query can be found here: http://www.nlm.nih.gov/bsd/pubmed_subsets/sysreviews_strategy.html).

**Table 4. Resources used for this experiment settings.**

| | **Terms Count** | **Documents Count** | **Topic** |
|---|---|---|---|
| PubMed Dental Journals | 182641 | 236767 | Dental |
| PubMed Nursing Journals | 74000 | 72494 | Nursing |
| PubMed Systematic Reviews | 219656 | 214252 | Clinical |
| BioNLP Corpus | 9483 | 908 | Biology |
| Reuters Corpus | 105675 | 694335 | News |
| ADR-Tweets Corpus | 6205 | 2008 | Drug |
| UMLS Definitions | 103933 | 188647 | Clinical |

We filtered out articles that do not have publicly available abstracts. Table 4 shows the number of documents in each corpus. We are also interested in evaluating additional corpora instead of only those generated from Pubmed. ASA uses all of the corpora matched with LSA as features to train the hybrid model. When using LSA independently for evaluation, outside ASA, only one of the corpora is used for each run. For the semantic similarity match step, we evaluated the following different settings:

1. Most similar concept returned by LSA using each of the corpora listed in Table 4.
2. Most similar concept returned by ASA.

*Synonym Normalization: Strict vs. Relaxed Evaluation*

For strict evaluation a correct prediction is when the predicted concept is exactly the same as the expected concept. On the other hand, in the relaxed evaluation mode, before calculating the evaluation metrics, we changed the predicted class to the expected class if the predicted class had any of these relationships in UMLS: "synonym," "is-a," "mapped-to" relations with the expected class. This means that if the system predicts a concept which is, for example, the synonym, child or parent of the expected concept, we considered it as a true positive. Considering the size of UMLS graph, we only did this normalization by distance of 2. Meaning that if a concept "a" has is-a relation with a concept "b," and the concept "b" has a "mapped-to" relation with a concept "c," the concept "a" and "c" would be considered the same for the evaluation purpose.

**Section 2.3 Case Study 2: Gene Functions Extraction with Semantic Relatedness**

As discussed in the first chapter, finding gene functions discussed in literature is crucial to genomic information extraction. The BioCreative IV GO workshop [43] aims to automate

gene function curation though computational methods. The goal of the shared task is to find the related gene functions (GO terms) in a set of genes discussed in an article. List of research publications along with their gene annotations are given to the system, and the expected output is the list of GO terms for gene functions discussed in the literature for each gene.

Here, we proposed two methods based on distributional semantic similarity that can be easily applied for different types of texts and ontologies. Both methods are based on a semantic relatedness function, and the system was evaluated with two measure of semantic relatedness: 1. Latent Semantic Analysis (LSA); and 2. The newly proposed hybrid method (ASA). This technique requires no engineered feature and it is very interesting to see how the results compare to existing machine-learning-based methods. If the performance is on par with the supervised methods, then we can benefit from unsupervised technique to leverage the performance of the supervised methods.

None of the previous work in BioCreative used semantic similarity methods including vector- or graph-based. The proposed technique here can be completely unsupervised if the semantic relatedness does not require any training (e.g. LSA); this characteristic makes the method unlikely to over-fit the dataset and very generalizable to the extraction of any major concepts mentioned in a document. The proposed method using LSA achieved the 3$^{rd}$ highest F-measure amongst 7 participants in the shared task. The proposed method is based on semantic similarity of sentences to GO terms and is capable of utilizing any semantic relatedness method.

In this work two semantic relatedness techniques were compared: LSA and ASA. The original paper [107] was based on LSA and was prepared for BioCreative IV GO shared task (Subtask b) [43,108]. The next sections discuss the general extraction pipeline followed by LSA and ASA specific preprocesses.

**Extraction Pipeline**

The extraction pipeline is the general process and remains constant regardless of which Measure of Semantic Relatedness (MSR) method was used. Figure 4 shows the overall flow of the proposed method. After MSR preprocessing was finished (discussed for each MSRs in the next sections), the objective was to find whether or not a sentence is related to a gene. This was done by using lexical patterns and generalizing the sentence and gene symbol (e.g. removing the numbers and non-alphabetic characters). If "Sentence Gene Matcher" predicts that a sentence is related to a gene, then we calculate its semantic similarity to all GO terms using already generated semantic vectors. The articles are provided in BioC format [109] in which sentences, passages and the types of passages (heading, paragraph, etc.) are identified. The "Go Finder" module finds all related GO terms to the sentence using a heuristic logic and generates the triplet of sentence, gene and GO term. Finally, the shared task expected output format is generated by "BioC output generator."

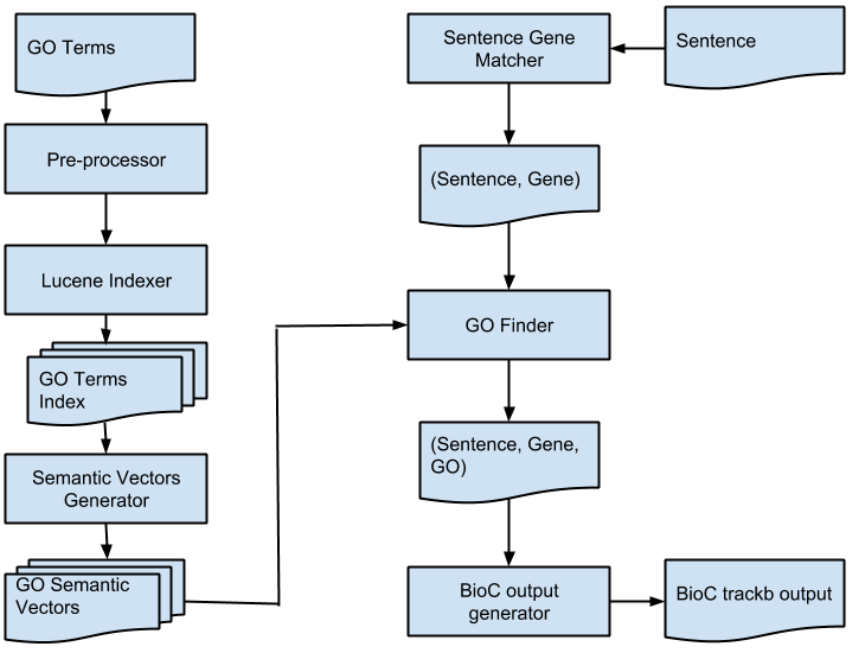


**Figure 4. High level flow of the proposed system. The left column shows the steps to create semantic vectors for each GO term. The right column displays the steps for finding GO terms in a document.**

GO Finder finds related GO terms for each sentence. We defined G as a set of top *m* GO terms with highest semantic similarity to the sentence. D is the set of top *n* GO terms with high similarity to the abstract of the related article. The following function returns top *k* similar GO terms for a given query:

$$TopSimilarGO(query, k) =$$

$$\{x | x \in GOTerms \ \wedge \ |\{y \ \in GOTerms \mid Sim(x, query) < Sim(y, query)\}| \ < k\}$$

And G and D sets are:

$$G(sentence) = TopSimilarGO(sentence, m)$$

$$D(abstract) = TopSimilarGO(abstract, n)$$

If a sentence is predicted to have the gene mention, the predicted GO terms for the sentence and gene are the conjunction of top similar GO terms to the sentence (set G) and top similar GO terms to the related abstract (set D):

$$GeneGO(gene, sentence, abstract)$$
$$= \{G(sentence)$$
$$\cap D(abstract)\}\ if\ HasGene(sentence, gene)\ \ else\ \{\}$$

A GO term with the highest semantic similarity to the sentence in the GeneGO set will be chosen as the final GO annotation for each gene in the sentence. For example if a sentence top *m*(=2) similar GO terms are {g5, g10} and the abstract top *n*(=5) GO terms are {g4, g8, g5, g2, g9}, then the final predicted GO terms for the sentence related to the gene will be {g5}. The tuning parameters *m* and *n* control precision and recall.

**Table 5. Number of sentences in the training set which were detected by "Sentence Gene Matcher" as relevant to a gene and annotated to have a gene function**. The different passage types are: "front" for the title of the article; "title_1" refers to section headings like "Introduction;" "title_2" is the section sub-headings that sometimes describes the specific topic/finding of the section; "title_3" and "title_4" are more deeper levels of section headings; "abstract" is the abstract content; "fig_title_caption" is the title of a figure caption and "fig_caption" is the caption of the figure; and "table_title_caption" is the caption of a table.

| Passage type | With Gene Function | Total | Percent |
|---|---|---|---|
| Front | 26 | 67 | 39% |
| title_2 | 149 | 797 | 19% |
| Abstract | 225 | 1253 | 18% |
| Paragraph | 1700 | 20703 | 8% |
| fig_title_caption | 17 | 412 | 4% |
| fig_caption | 99 | 6009 | 2% |
| table_title_caption | 0 | 47 | 0% |
| title_1, title_3, title_4 | 0 | 26 | 0% |

Table 5 summarizes the number of sentences in the training set which were detected by "Sentence Gene Matcher" as relevant to a gene and also annotated to have a gene function. The table shows that "abstract," "front" and "title2" sections of each document are the most important sections that can include gene function. The passage types appearing in Table 5 are exactly taken from the corpus. Table 6 shows an example for each passage types from publications in the train set. We found that the first sentences of paragraphs have information about GO terms, but including all sentences in a paragraph will significantly reduce the precision. Therefore, we limited searching for the gene functions to the mentioned sections of the article. We chose one set of values for m and n, for "Front," "Abstract" and "Title2" (mFAT, nFAT), and chose a different set for the first sentence of

the paragraphs (mParagraph, nParagraph). Figure 5 illustrates the process of generating output with an example. Next section shows the detail analysis of the impact of the tuning parameter on precision and recall.

**Table 6. Description of different passage types appeared in the corpus along with an example for each type**. Title_3 and Title_4 are very similar, but we maintained the naming from the corpus to keep it consistent with the data.

| Passage type | Description | Example |
|---|---|---|
| Front | The title of the document | Activation of ASK1, downstream MAPKK and MAPK isoforms during cardiac ischaemia |
| Abstract | The content of abstract section of the article | p38 MAPK is activated potently during cardiac ischaemia, although the precise mechanism by which it is activated is unclear. We used the isolated perfused rat heart… |
| Title_1 | Section title | "Introduction", "Results", "Discussion" |
| Title_2 | Subsection title. (See the highlighted part in the example.) | Results<br>…<br>**Nuclear Translocation of Fussel through Medea**<br>The presence of a SMAD binding domain in the Fuss Protein… |
| Title_3 | Subsubsection title. An inline heading which appears at the beginning of a paragraph. (See the highlighted part in the example.) | Materials and Methods<br>RNA interference by feeding The RNAi feeding vectors were either made in our laboratory using<br>GC analysis A mixed population of well-fed worms were washed … |
| Title_4 | An inline sub-heading which appears at the beginning of a paragraph. (See the highlighted part in the example.) | Materials The super8XTOPFLASH (superTOP) reporter construct containing eight Lef/TCF…<br>Image Analysis Western blots were analyzed using the ImageJ program, and band volumes were quantitated. |

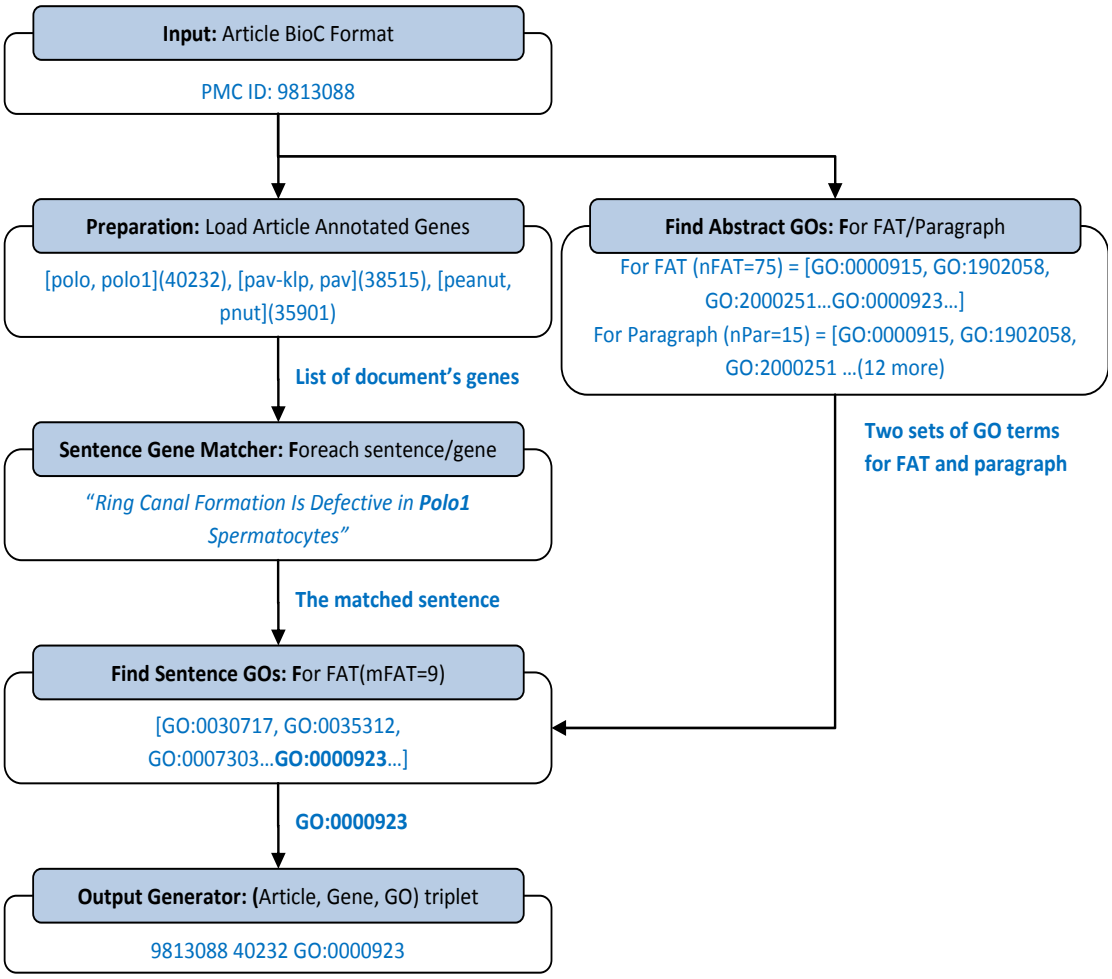


**Figure 5. Process of finding GO terms for each gene in a given document by an example. The example sentence category is "front_2" (FAT sections). Exception the value for *n* and *m* parameters, the process is the same as FAT for sentences in paragraphs.**

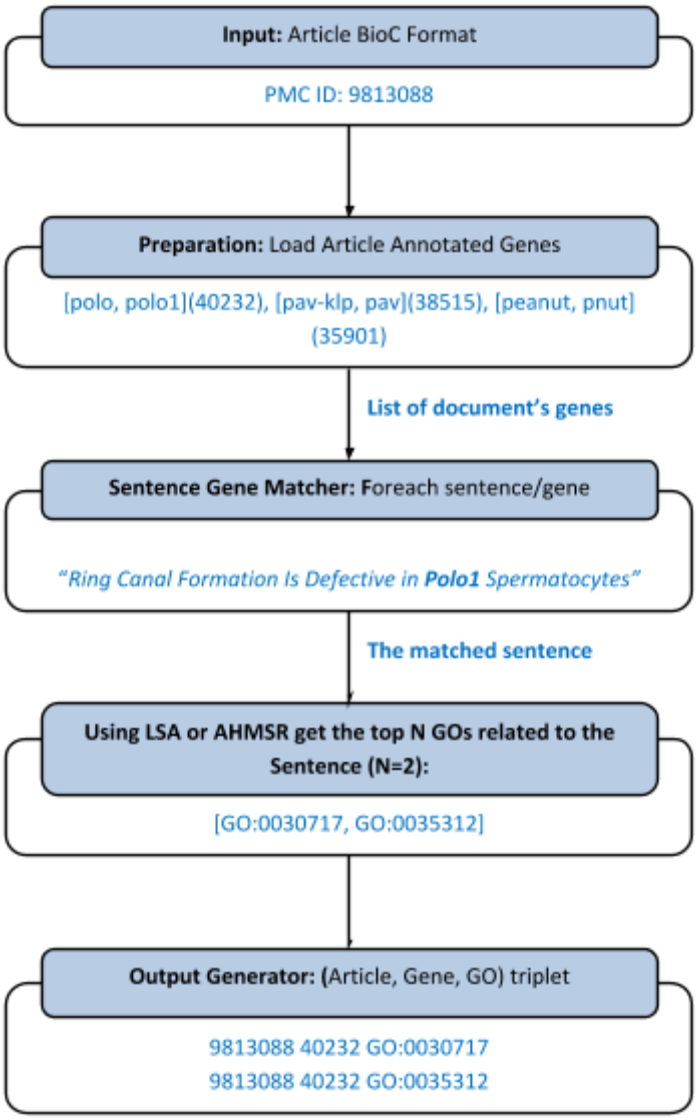


**Figure 6. Steps involved in gene function extraction without using the intersection technique (NoIntersection)**.

**MSR Method: LSA**

For LSA [15] implementation, Semantic Vectors Package [110] is used with random indexing [111]. GO terms' semantic vectors were created based on GO names and definitions in GO; one semantic vector was created for each term in the ontology. Stop-

words were removed from GO name / definition and they are generalized by Porter stemming [112]. The semantic relatedness of a phrase to a GO term was calculated by cosine distance of the phrase representative vector and the GO terms semantic vector. The phrase representative vector is the average vector of all the terms' vectors in the phrase.

**MSR Method: ASA**

The previous section explains a technique how to use LSA to find gene functions mentioned in a document using intersection of the sentences related GO terms and abstract related GO terms. For evaluating ASA, in addition to use the intersection method, the simple method which returns the most similar GO term to each sentence that possibly matches with a gene was evaluated (referred to as NoIntersection). ASA was trained on the training and development sets and evaluated on the testing set.

ASA training examples were created from the annotation in development and training sets. For each gene function annotation of a sentence, a training example was generated for a pair of the sentence content and the annotated GO term name. The expected relatedness score for the positive training examples is set to 100. For each positive example [10] negative examples were generated by randomly selecting different GO terms. In this experiment only LSA as MSR was used and GoTerm, GoTermDefinition, GoTermDefinitionSynonym, BioNLP, ADRTweets, PubmedDentalJournals, PubmedNursingJournals (described in Table 4) were used as resources. Figure 7 explains ASA training process, which was performed once using the training and development sets.

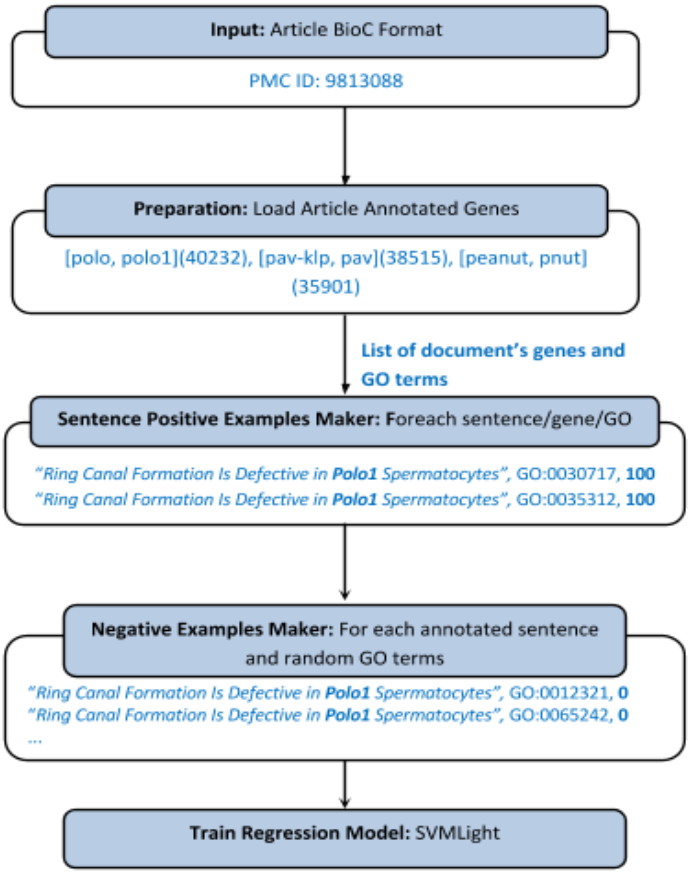


**Figure 7. Process of training ASA model with an example.**

In Summary, this chapter introduce a novel hybrid semantic relatedness which is capable of combining heterogeneous MSRs and resources. Two uses cases that use semantic relatedness as part of their solution were introduced. 0The next three chapters discuss the experiment setup and evaluation results for all methods described in this chapter. 0 discusses intrinsic evaluation of ASA and evaluates the ASA as an independent semantic relatedness function. 0 shows the effectiveness of ASA for ADR normalization problem. 0 presents the evaluation results for Gene function extraction and analyzes the impact of the tuning parameter on precision and recall.

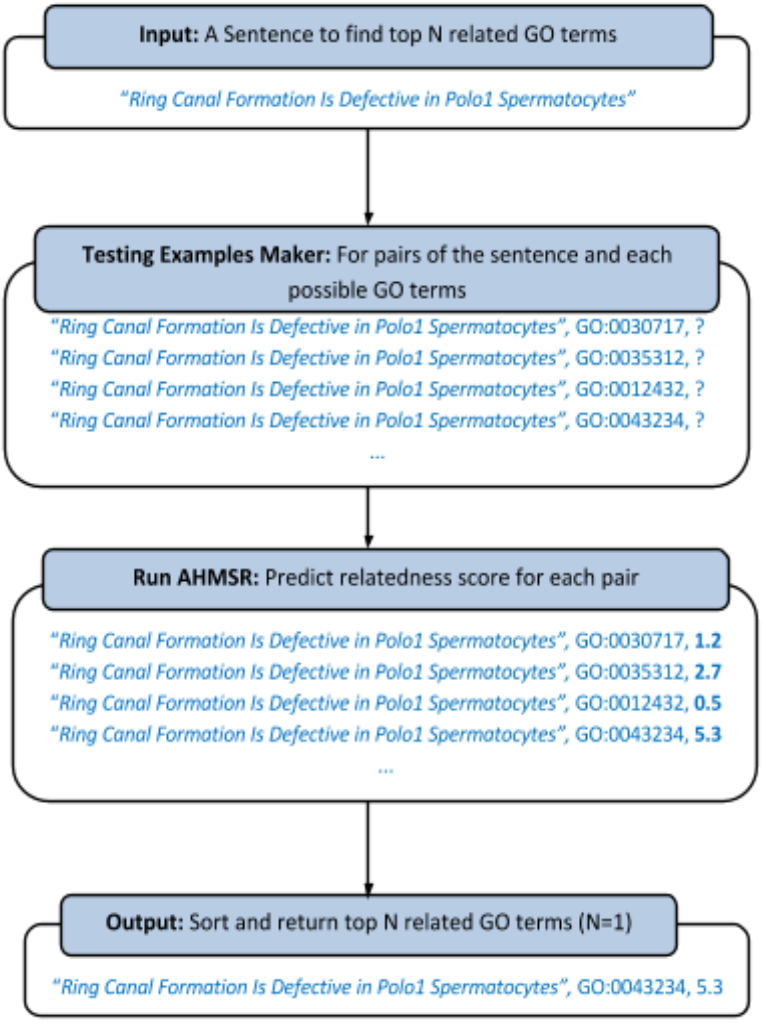


**Figure 8. ASA used to predict the top N related GO terms to a given sentence**.

# 3 INTRINSIC EVALUATION

Intrinsic evaluation aims to evaluate a natural language processing technique in an isolated manner disconnected from other tools or components. This contrasts with extrinsic evaluation which attempts to assess the performance change in a complex system with multiple components when a single component of the system changes. This chapter focuses on the intrinsic evaluation of ASA followed by two chapters covering extrinsic evaluations of the proposed system in ADR normalization and gene function extraction.

## Experimental Setup

Intrinsic evaluation directly compares the proposed method with experts' ratings. This evaluation is good for evaluating how similar the MSR is to the experts. The correlation between human judgment and system output is usually used to evaluate an MSR. We use three textual corpora created from three different queries on PubMed (provided as special queries: http://www.nlm.nih.gov/bsd/special_queries.html): Dental Journal[6], Nursing Journals[7] and Systematic Reviews[8]. We limited our search results to those articles with available abstracts. In addition we use Yahoo! search with PMI. Table 7 shows the list of resources and MSRs that were used for this study. While these resources might not reach the best possible results, we chose this list of resources to provide a comparison baseline for different MSR techniques. In order to evaluate our different techniques, we use the UMN Semantic Relatedness and Similarity (UMNSRS) [25] benchmark. UMNSRS is the most suitable evaluation set for the proposed hybrid method since it includes enough pairs

[6] PubMed query: "(jsubsetd[text])"
[7] PubMed query: "(jsubsetn[text])"
[8] The query can be found here:
http://www.nlm.nih.gov/bsd/pubmed_subsets/sysreviews_strategy.html

to train the regression model effectively. The benchmark consists of 566 pairs rated for similarity and 587 pairs rated for relatedness by eight medical residents. In order to compare results to the previous works, we reported both Pearson product-moment correlation coefficient (R) and Spearman's rank correlation coefficient (ρ). We use Fisher r-to-z transformation to assess the significance of differences in R values. For evaluating ASA, since it requires training set, we applied 2-fold cross-validation for generating output on the benchmark. This means that we trained on the first half and evaluate on the second half of the UMN set and repeated the training on the second half and testing on the first half.

Equation 8 Pearson Correlation

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

Equation 9 Spearman correlation

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)}.$$

**Table 7. Resources and matched MSRs used in this experiment setting**. LSA and deep learning contextual vectors are matched with PubMed corpora. PMI is matched with all resources.

| **Resources** | **Terms Count** | **Documents Count** | **LSA** | **PMI** | **NN Skip-gram Vectors** |
|---|---|---|---|---|---|
| PubMed Dental Journals | 182641 | 236767 | √ | √ | √ |
| PubMed Nursing Journals | 74000 | 72494 | √ | √ | √ |
| PubMed Systematic Reviews | 219656 | 214252 | √ | √ | √ |
| YahooAsCorpus | - | 15*10^6 | - | √ | - |

***Gold Standard***

The semantic relatedness gold standard can be different in each domain. Better gold standards are those that summarize a significant number of experts' opinions about the relatedness of two concepts in their field of expertise. For example, "bridge" and "edentulism" can sound relevant to dentists while they seem quite irrelevant to civil engineers. This highlights the importance of selecting the proper level and kind of expertise for each problem in order to create the gold standard.

There cannot be a general corpus which works for every specialty domain, although we can have one gold standard based on general knowledge scoring of general English concepts. Even though, experts might score the general concepts differently, this general ground truth can help solve more generalized problems by encoding the common sense of humans (for example "sea" and "ship"). Pakhomov et al. [113] studied the process of the gold standard creation for semantic similarity and tried to propose a framework for semantic relatedness reference standards in the clinical domain. They wanted to address the problem of reproducing semantic analysis results, and proposed a set of tools and flows to standardize the semantic relatedness process. For evaluating semantic relatedness, different evaluation methods exist.

We can evaluate an MSR by calculating mean square error from experts' ratings. Another evaluation method is to check how an MSR function can predict existing relations between concepts in an ontology. For example, concepts that are connected via a parent-child relation in an ontology are supposed to be strongly related; therefore the MSR should return high value for them. In addition to evaluating an MSR alone, we can evaluate its effect when it is used inside other systems like relationship classifiers or ontology mappers. For

example, we can measure a system's performance improvement after using a semantic relatedness score as a feature inside a machine learning-based link classifier.

**Results**

Table 8 shows the correlations of the outputs of each setting with human judgment. ASA outperforms all MSR kernels matched with any of the resources. The next highest correlation is achieved by using LSA with PubMed Systematic Reviews corpus. The PubMed Dental Journals corpus, with similar size to the Systematic Reviews set, achieved the lowest results. Since the concept pairs in UMN are general clinical concepts, this shows that relatedness of corpus to the concepts is more important than size of corpus. PubMed Systematic Reviews corpus yields better results than Dental and Nursing Journals consistently across all MSRs. The percentages of non-zero values returned by a method are shown in Table 8. Since we set the default value of "0" for a MSR when a word is not found in the resource, the non-zero percentage shows how much of the evaluation set is covered by the settings. ASA yields the highest possible completeness by returning a non-zero value for every pair in the evaluation set. PMI returns values for a fewer pairs in comparison with LSA, but it yields better correlation for most of the corpora except for the Systematic Reviews corpus. PMI outperforms skip-gram model for all corpora. LSA is only outperforms skip-gram when using PubMed Systematic Reviews corpus. The skip-gram model produces less non-zero output in compare to LSA, which can be one of the causes for the lower results than LSA.

**Table 8. Performance of each individual MSR matched with a resource on two evaluation set**. UMN relatedness and similarity. The last row shows results for the proposed hybrid system (ASA). "%Non-zero" column shows what percentage of evaluations set was assigned value not equal to "0" form the system.

| | **UMN similarity** | | | **UMN relatedness** | | |
|---|---|---|---|---|---|---|
| | **R** | **Spearman** | **% Non-zero** | **R** | **Spearman** | **% Non-zero** |
| LSA-PubMedDentalJournals | 0.09 | -0.01 | 61.30% | 0.05 | -0.03 | 57.92% |
| LSA-PubMedNursingJournals | 0.10 | 0.00 | 47.52% | 0.08 | 0.04 | 44.97% |
| LSA-PubMedSystematicReviews | *0.45* | *0.42* | 71.02% | *0.43* | *0.45* | 67.97% |
| Word2Vec—PubMedDentalJournals | 0.12 | 0.09 | 37.80% | 0.16 | 0.14 | 34.07% |
| Word2Vec – PubMedNursingJournals | 0.09 | 0.09 | 24.91% | 0.16 | 0.17 | 23.68% |
| Word2Vec – PubMedSystematicReviews | 0.29 | 0.21 | 53.00% | 0.30 | 0.26 | 50.09% |
| PMI-PubMedDentalJournals | 0.35 | 0.34 | 13.95% | 0.36 | 0.37 | 13.11% |
| PMI-PubMedNursingJournals | 0.32 | 0.29 | 10.95% | 0.35 | 0.37 | 10.73% |
| PMI-PubMedSystematicReviews | 0.40 | 0.36 | 27.56% | 0.39 | 0.42 | 26.40% |
| PMI-YahooAsCorpus | 0.32 | 0.32 | 99.29% | 0.21 | 0.21 | 99.29% |
| ASA | **0.53** | **0.53** | 100% | **0.49** | **0.51** | 100% |

Figure 9 shows scatter diagrams for the output of two settings: LSA-PubMedSystematicReviews (Figure 9.a) and ASA (Figure 9.b) on UMN similarity. The diagrams shows that ASA provides a consistent similarity for most of the pairs, while LSA-PubMedSystematicReviews output does not show a clear linear correlation with human

judgments (R=0.43). LSA-PubMedSystematicReviews could find vectors for 67.97% of the evaluation set, which can be the main reason that the scatter diagram is dense around horizontal axis. ASA benefits from combining resources to return a value for all of the pairs in the evaluation set. Namely the use of Yahoo! search as a resource enables the system to find at least one predictor (PMI-YahooAsCorpus) for every pair because of the huge size of the Yahoo! corpus. Similarly, Figure 10 shows scatter diagrams of the top two systems (LSA- PubMedSystematicReviews and ASA) for UMN relatedness set. Comparing the ASA output on relatedness to similarity set, the system seems to have more divergence at higher value for relatedness and it gives more consistent results for similarity set. Interestingly we found that skip-gram's word embedding on PubMedSystematicReviews performed significantly worse than LSA on the same corpus (p=0.0054).

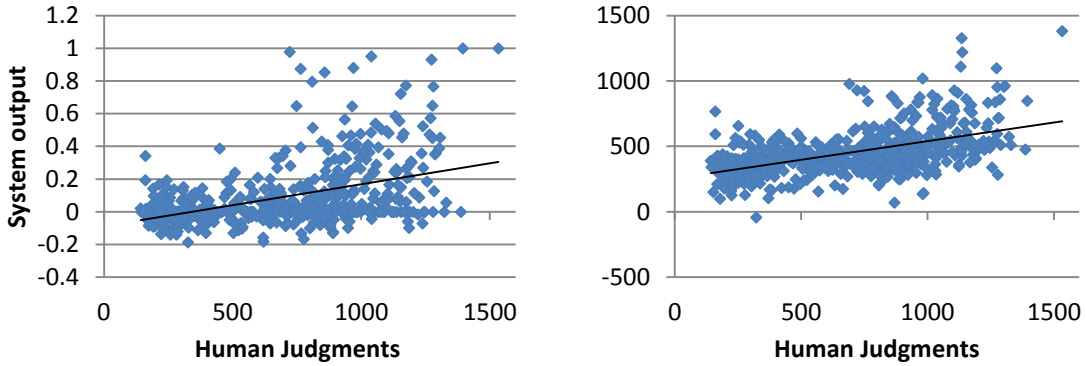


**Figure 9. Top two system's outputs on UMN similarity set. (a) The left diagram shows output of LSA using PubMedSystematicReviews corpus; and (b) the right diagram shows output of ASA system. The line is the linear trend line.**

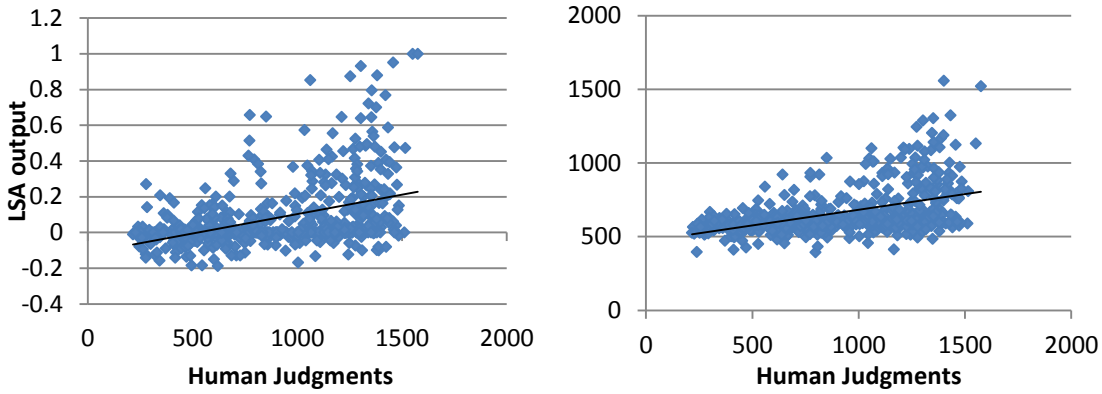


**Figure 10. Top two system's outputs on UMN relatedness set. (a) The left diagram shows output of LSA using PubMedSystematicReviews corpus; and (b) the right diagram shows output of ASA system. The line is the linear trend line.**

## Discussion

From the previous experiments on UMNSRS benchmark, Garla *et al.* [114] reported the highest Spearman correlation of 0.46 and 0.39 for UMN similarity and relatedness sets respectively. They showed that their method outperforms the vectors generated from 500,000 Electronic Medical Records (EMR) inpatient reports. Our experiment shows that LSA on Systematic Reviews corpus yields correlation of 0.42 and 0.45 for UMN similarity and relatedness sets. However the vector model correlation on similarity set is 0.04 lower than the Garla's method; but in relatedness set, LSA vectors outperforms by 0.06. This highlights the importance of the corpus content used for generating term vectors. ASA outperforms Garla's method by 0.07 (0.53 vs. 0.46) and 0.12 (0.51 vs. 0.39) for similarity and relatedness sets respectively. However, the words in the benchmark can be missing in the resource used to generate term vectors. This will cause the method to return a default value for the word-pair (0 in this experiment) worsening the correlation. The proposed hybrid method will reduce the chance of missing words by combining various resources. ASA is not a domain-specific metric and can adapt to various domains using the available domain-specific knowledge-bases and corpora.

I expected to get similar improvement in other domains (Biological and General English); but due to small size of similarity sets, we could not complete training the regression model for other domains. For example for general English concepts, the Rubenstein and Goodenough set contains only 65 word pairs (rated by 51 human subjects) [115] and Miller and Charles contains 30 word pairs (rated by 38 human subjects) [116].

**Limitations**

We recognized several limitations with this work. The LSA method performance is highly dependent on the quality and size of the corpus, and it is hurt greatly by missing terms in the corpus. We expect the LSA result to be improved by including larger, more relevant textual resources. In addition we only considered two traditional statistical and vector-based methods; future work will include performing experiments on other graph-based and information content techniques. One of the shortcomings of the proposed method is that it requires a big enough training set to create the regression model. This will likely cause the model not to perform well on other smaller benchmarks.

In Summary, this chapter evaluated the proposed semantic relatedness against expert ratings and showed improvement over each individual semantic relatedness technique. The next two chapters evaluate the effect of replacing existing semantic relatedness technique in an existing solution pipeline with the proposed semantic relatedness technique.

# 4 CASE STUDY 1 RESULTS: NAMED ENTITY NORMALIZATION

## Experiment Setup

Extrinsic evaluation is about measuring the changes when the new algorithm is used inside another system. In our case, it is about finding out how effective the new semantic similarity model can be when it is used for normalizing Adverse Drug Reaction (ADR) from patients' tweets about a drug. As discussed in the method section, a system was proposed to use ASA for the normalization task. In this chapter the proposed system with a setting which only use LSA is compared to when LSA is used independently.

The following empirical factors were used for evaluating the ADR normalization results: Precision, Recall and F-Measure. From the perspective of evaluation, each UMLS concept is considered to be a class. The final precision, recall, and F-measure were calculated as the micro-average of all the classes. For each class true positive (TP), false positive (FP), true negative (TN) and false negative (FN) are defined as below: TP is when the expected class is equal to the predicted class and the evaluated class. FP is when the predicted class is equal to the evaluated class but not equal to the expected class. FN is when the expected class is equal to the evaluated class but the predicted class is not equal to the expected class. TN is when both predicted and expected class are not equal to the evaluated class. Table 9 illustrates an example for the evaluation strategy. The micro-averaged precision and recall were calculated using the following formula:

$$Precision = \left(\sum_{c \in Classes} TP_c\right) / \left(\sum_{c \in Classes} (TP_c + FP_c)\right)$$

Equation 10. Precision for Normalization

$$Recall = (\sum_{c \in Classes} TP_c) / (\sum_{c \in Classes} (TP_c + FN_c))$$

Equation 11. Recall for Normalization

F-measure is the harmonic average of the micro-averaged precision and recall:

$$F - measure = \frac{2 * Precision * Recall}{(Precision + Recall)}$$

Equation 12. F-Measure

**Table 9. This table illustrates an example situation to explain the evaluation technique.**

| Mention | Expected Class | Predicted Class | Evaluated class | |
|---|---|---|---|---|
| | | | Class1 | Class2 |
| M1 | Class1 | Class1 | TP | TN |
| M2 | Class1 | Class2 | FN | FP |
| M3 | Class2 | Class1 | FP | FN |
| M4 | Class2 | Class2 | TN | TP |

**Results**

Table 10 shows the results for syntactic matcher, LSA, using different corpora and the proposed hybrid model. ASA yielded the best F-measure of 62.37 and the best recall of 50.20. The next best precision after syntactic match was achieved by LSA with UMLS definitions corpus. Among LSA with various corpus, ADR-Tweets resulted in the best F-measure. In the investigated normalization problem, ADR-Tweets corpus yielded the best

performance for LSA method. Syntactic matcher had the highest precision which was expected. Adding LSA-ADR-Tweets matcher on top of syntactical matcher decreased the precision but increased the recall resulting in significantly higher F-measure. Using ASA instead of LSA, decreased the precision slightly more than LSA but the gain on recall was higher and resulted in a higher F-measure. We used the relaxed evaluation method in all of the reported results.

**Table 10. Shows the results of the proposed pipeline using relaxed evaluation technique.**

| | Precision | Recall | F-Measure |
|---|---|---|---|
| Syntactical | **88.0** | 35.7 | 50.8 |
| LSA-PubM-Dental | 83.6 | 38.2 | 52.4 |
| LSA-PubM-Nursing | 83.1 | 38.6 | 52.7 |
| LSA-UMLS-Defs | **86.5** | 40.3 | 55.0 |
| LSA-Reuters | 81.5 | 44.9 | 57.9 |
| LSA-PubM-Systematic | 83.6 | 44.4 | 58.0 |
| LSA-ADR-Tweets | 84.6 | 47.7 | 61.0 |
| ASA | 82.3 | **50.2** | **62.4** |

**Discussion**

Figure 11 shows false positive and true positive sources. Semantic match generates most of false positives followed by exact match. Exact match returns the majority of true positives followed by a semantic match. As expected, the syntactic matching component, when applied by itself, obtains high precision but very low recall. Searching for exact match in definition helps to find alternative representation of the concept. For instance "urge to vomit" is normalized correctly to "c0027497-Nausea" when we searched the

definition of “c0027497”: “unpleasant sensation in the stomach usually accompanied by the urge to vomit.” Exact match fails when the words in a phrase are expressed in complex orders and another concept matches exactly with the annotated phrase. For instance, in the following tweet: "dreams have taken a terrifying turn," and "dreams" is annotated as "c0857051-bad dreams" but exact match matches the phrase with "c0028084-dreams."

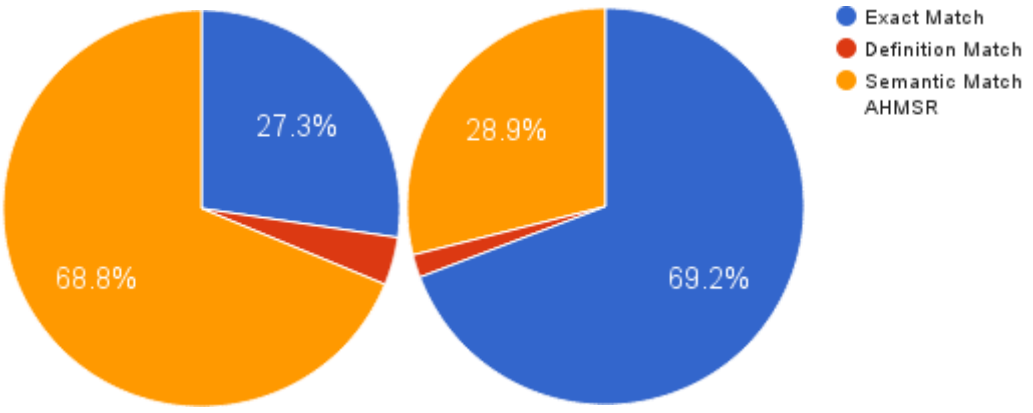


**Figure 11. Source of correct and wrong predictions. Left chart shows percentage of false positives from each component and the right chart shows true positive percentages.**

In contrast, semantic methods are designed to compute estimates of similarity, and match concepts that are not necessarily the same, but are similar. As such, they are expected to have high recall. In our experiments, the semantic matchers LSA and ASA have the highest number of false positives but yield higher recalls than syntactic match. This was expected since most of hard to normalize concepts reach the semantic matchers modules. Most of the errors are caused by concepts with very similar meanings. For example, "anti-depressant" in a tweet is tagged as "c0011570-mental depression," but LSA returns "c0005586-manic depression" as the most similar concept. Table 11 shows examples of correct and incorrect predictions by ASA. The hybrid model is very good at normalizing

when the same word is represented in a different variation (“antidepressant” vs. “depression”) or match similar words which appear frequently in corpora (“fewer” vs. “loss”, “increase” vs. “gain”). But on the other hand, ASA performance is limited to the information in the provided resources and MSR technique. Since in this experiment we only used LSA, ASA would behave solely based on co-occurrences of terms in the resources. If there is not enough co-occurrences of two words in the provided resources then we expect to have a very low similarity of the terms. In addition to using larger corpora, adding more diverse techniques which can leverage other resource types (such as graph-based techniques) can significantly boost this limit.

**Table 11. Lists some example of correct and incorrect predictions by ASA. The first four rows are correct predictions followed by three rows of incorrect predictions. The last row is correct by using relaxed evaluation.**

| Annotated Phrase | Expected | Predicted |
|---|---|---|
| Antidepressant | c0011570-Depression | c0011570 |
| increase my weight | c0043094-Weight gain | c0043094 |
| gain so much weight | c0043094-Weight gain | c0043094 |
| fewer hours sleep | c0235161-Sleep loss | c0235161 |
| feel like need to throw up | c0027497-Nausea | c0917799-Hypersomnia |
| just eat, and eat | c0232461-Apetite increase | c0015672-Fatigue |
| falling asleep every day | c0541854-Daytime sleepiness | c0917801-Insomnia |
| it's 4:30am. at this point ima just throw out a big "f*** you" | c0917801-Insomnia | c0917799-Hypersomnia |

In Summary, we showed using the proposed hybrid model instead of traditional semantic relatedness methods, improves the results in adverse drug reaction normalization. The next chapter seeks to answer if the proposed hybrid model enhance the solution for gene function extraction.

# 5 CASE STUDY 2 RESULTS: GENE FUNCTION EXTRACTION

## Experiment Setup

When LSA is used as the semantic relatedness method, the technique becomes completely unsupervised; hence no training is involved. However we still need to tune various parameters of the system to achieve the best f-measure. The dev-set is used for this purpose to find optimal values for tuning parameters. Then the results are reported on the training and testing sets. On the other hand, ASA requires training to generate the regression model. The training set is used to train the model and results are reported on the testing set. Development set is used to tune the parameters for ASA.

## Tuning Parameters

To achieve the highest F-measure, the tuning parameters (*m* and *n*) need to be adjusted accordingly. We use two sets of values for *m* and *n*; one set for the first sentence of each paragraph (*mParagraph* and *nParagraph*) and another for FAT passage types ($m_{FAT}$ and *nFAT*). To find the best tuning parameters, we evaluated the system with different values for a particular parameter while values of other parameters were constant. The experiment was repeated for all four parameters. Figure 12 shows variation of performance for LSA technique when tuning parameters change. Overall when parameters increase, precision increases and recall decreases. The goal is to find the values which yield maximum F-measure. Figure 12-a depicts precision, recall and F-measure change in respect to *mFAT* changes. As *mFAT* increases, precision declines and recall increases. For LSA technique the maximum F-measure is achieved for *mFAT=9*. Therefore we assigned *mFAT* to 9, and attempted to find the best value for *mParagraph*. Figure 12-b shows the change of performance based on the change of *mParagraph* and best result achieved for

*mParagraph=15*. Figure 12-c shows variation of performance when *nFAT* varies and Figure 12-d shows performance change when *nParagraph* is changed while other parameters are constant. The best F-measure of 0.294 is achieved for *mFAT*=9, *mParagraph=2, nParagraph=15 and nFAT=75.*

When mParagraph varies, the change in F-measure is not as significant as when mFAT varies. In addition, recall is almost constant for mFAT >2. This shows that considering more than 2 GO terms for each sentence in FAT sections does not help us much and can only decrease the precision. On the other hand, only one top GO term for the first sentence of each paragraph gives the maximum boost to the recall.

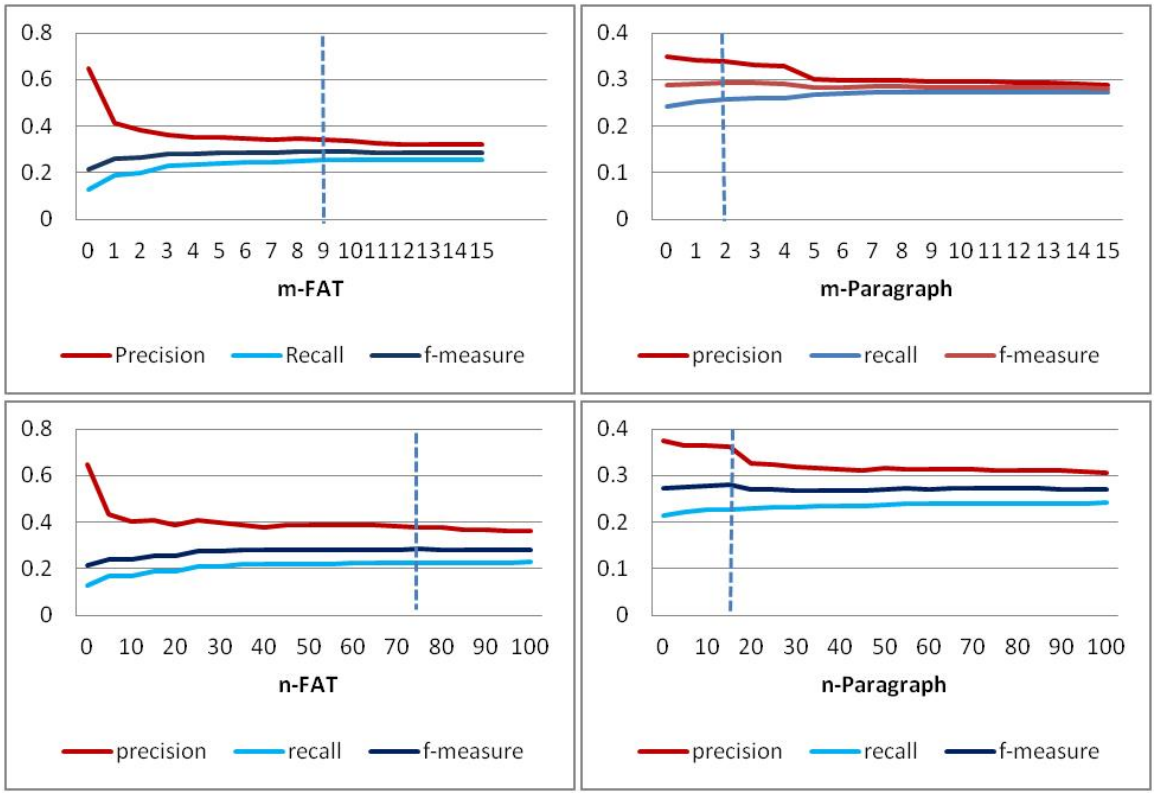


**Figure 12**. a) Top-left diagram depicts precision, recall and F-measure change in respect to *mFAT* ("Front", "Abstract" and "Title") changes when other parameters have constant values (*mParagraph=1, nFAT=100, nParagraph=15*). b) Top-right diagram shows the change of performance based on changes of *mParagraph* when *mFAT=9, nFAT=100, nParagraph=15*. c) Bottom-left diagram shows the change of performance when *nFAT*

varies and *mFAT=3, mParagraph=1, nParagraph=15*. d) Bottom-right diagram shows the change of performance when *nParagraph* varies and *mFAT=3, mParagraph=1, nFAT=100*.

The tuning parameters should be readjusted with new values when ASA is used instead of LSA. Following the same tuning approach for ASA, Figure 13 and Figure 14 show variation of performance on development set with different values for parameters. The result on development set (f-measure=0.30) achieved with the following parameters values: m-FAT = 9, m-Par=2, n-FAT=0 and n-Par=40.

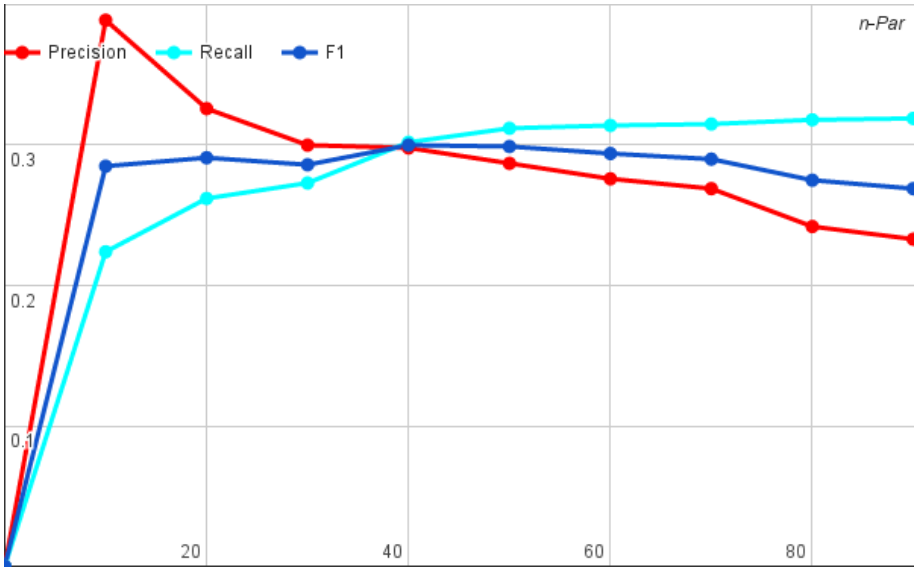


**Figure 13. ASA performance on development set. When n-Par varies and other parameters have constant values: m-FAT = 9, m-Par=2 and n-FAT=0.**

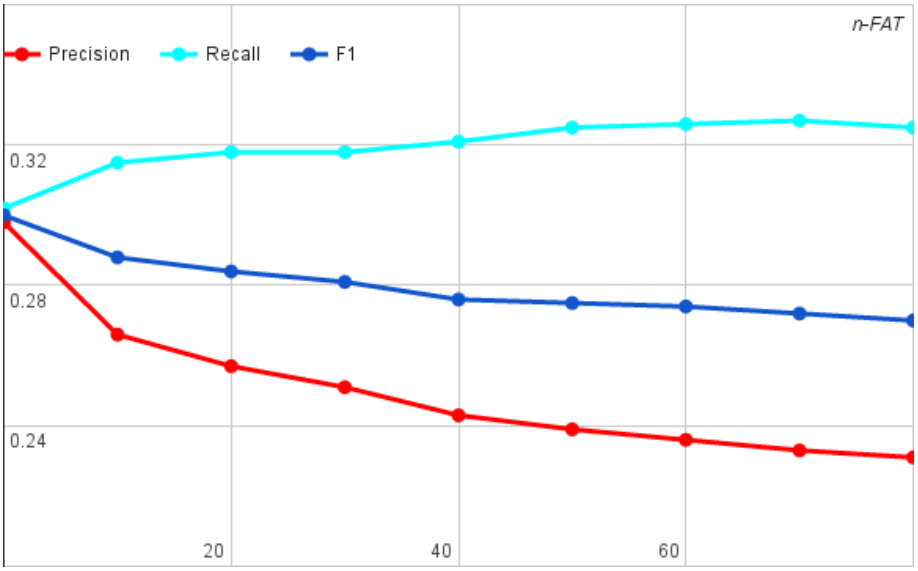


**Figure 14. ASA performance on development set. When n-FAT varies and other parameters have constant values: m-FAT = 9, m-Par=2 and n-Par=40.**

**Results**

Having the tuned parameters, we compared the performance of the proposed intersection approach to alternative systems (without intersection algorithm or limit on section types). In addition, we compared the contribution of the first and the last sentences of paragraphs. Table 12 shows the performance of different settings. The first experiment tested how much the intersection approach improved the results in comparison to just finding semantic similarity of each sentence in the LSA approach. The first four rows in Table 12 were achieved by not using intersection but simply the most similar GO term to each sentence. The last five rows in Table 12 were achieved using the intersection method. The best recall (0.518) was achieved by not using intersection and not limiting scope to any specific parts of the document; however the precision was very low.

Limiting the scope to paragraph and FAT improved the precision slightly (+0.009) and decreased recall (-0.020). Similarly including only Paragraph section improved precision

and reduced recall a little more (+0.010 precision, -0.025 recall). When only the FAT section was included, the precision increased significantly and recall also dropped sharply (+0.199 precision, -0.246 recall). This yields a higher F-measure than including paragraph or all sections. In short, when we limit the scope, the precision increases and recall decreases. We see the same pattern with intersection approach but precision remains high in comparison with no-intersection approach. When we compared intersection and no-intersection approaches including all sections (Table 12, row 1 and row 5), it showed that intersection reduced recall by 0.213 but increased the precision by 0.186. In another experiment we found that limiting search to first sentence of paragraph sections can improve the precision significantly. The last four rows of Table 12 compare the performance when different parts of the paragraph are included; they show that including the first sentence yields the best F-measure and precision.

In Table 13, we compared four settings for creating semantic vectors: 1) using only the GO terms; 2) using GO term and definition; 3) using GO term and synonym; and 4) using GO term, definition and synonym. Using only terms to create vectors achieved the best results. This may be mainly due to the similarity of GO terms and more description inclusion causes the vector to easily return incorrect GO term with higher similarity.

**Table 12. Performance of different settings on dev-set for LSA approach**. For intersection approach the tuning parameter values are mFAT=9, mParagraph=2, nParagraph=15 and nFAT=75. Random Index algorithm random function's seed was fixed to "1234."

| | *Precision* | *Recall* | *F-measure* |
|---|---|---|---|
| LSA-No intersection/All sections included | 0.082 | **0.518** | 0.141 |
| LSA-No intersection/Paragraph+FAT | 0.091 | 0.498 | 0.155 |
| LSA-No intersection/Paragraph | 0.092 | 0.493 | 0.155 |
| LSA-No intersection/FAT | 0.281 | 0.272 | 0.276 |
| LSA-Intersection/All section | 0.268 | 0.305 | 0.285 |
| LSA-Intersection/Paragraph last sentence+FAT | 0.346 | 0.245 | 0.287 |
| LSA-Intersection/Paragraph all sentences+FAT | 0.316 | 0.278 | 0.296 |
| LSA-Intersection/Paragraph last and first sentences+FAT | 0.348 | 0.261 | **0.299** |
| LSA-Intersection/Paragraph first sentence+FAT | **0.366** | 0.252 | 0.298 |

**Table 13. Comparing four settings for creating semantic vectors. 1) Using only the GO terms, 2) using GO term and definition, 3) using GO term and synonym, and 4) using GO term, definition and synonym. For all experiments in this table, FAT and Paragraph (only first sentence) sections are considered.**

| | *Precision* | *Recall* | *F-measure* |
|---|---|---|---|
| Create vectors with GO terms only | **0.366** | **0.252** | **0.298** |
| Create vectors with GO terms+definitions | 0.247 | 0.229 | 0.238 |
| Create vectors with GO terms+definitions+ synonyms | 0.227 | 0.196 | 0.210 |
| Create vectors with GO terms+synonym | 0.197 | 0.189 | 0.193 |

LSA and ASA results on the testing set are reported in Table 14 and Table 15 for hierarchical and exact match evaluations respectively. ASA yields higher f-measure than LSA in all settings. In NoIntersection technique ASA precision is slightly lower than LSA but the f-measure is higher due to the increase in recall. The different is more clear when we use exact match evaluation. In exact match ASA outperforms LSA in precision, recall and f-measure in all settings.

**Table 14. Comparing the hierarchical evaluation of ASA and LSA on the testing set in various settings**.

| | *Precision* | *Recall* | *F-measure* |
|---|---|---|---|
| LSA-NoIntersection/Paragraph last and first sentences+FAT | 0.101 | **0.583** | 0.172 |
| ASA-NoIntersection/Paragraph last and first sentences+FAT | **0.129** | 0.517 | **0.207** |

| | | | |
|---|---|---|---|
| LSA-Intersection/ Paragraph last and first sentences+FAT | 0.202 | **0.393** | 0.267 |
| ASA-Intersection/Paragraph last and first sentences+FAT | **0.23** | 0.384 | **0.288** |

**Table 15. Comparing the exact match evaluation of ASA and LSA on the testing set in various settings.**

| | *Precision* | *Recall* | *F-measure* |
|---|---|---|---|
| LSA-NoIntersection/Paragraph last and first sentences+FAT | 0.012 | 0.076 | 0.02 |
| ASA-NoIntersection/Paragraph last and first sentences+FAT | **0.038** | **0.247** | **0.066** |
| LSA-Intersection/Paragraph last and first sentences+FAT | 0.045 | 0.059 | 0.051 |
| ASA-Intersection/Paragraph last and first sentences+FAT | **0.122** | **0.177** | **0.144** |

## DISCUSSION & SUMMARY

We proposed a supervised and an unsupervised approach to extract gene functions from documents. The unsupervised approach only uses GO terms' names for creating semantic vectors. We attempted using GO terms description but it did not help. Using a more fine-tuned vocabulary set for each GO term may result in more accurate vectors and may increase the performance of this method. In addition, using term-term semantic similarity for expanding sentence terms can be evaluated. We used annotations for finding the

important passage types, evaluating the method, and finding the best settings for the parameters. The main advantage of using unsupervised open IE technique is that it can easily be generalized and applied to similar relation extraction problems. ASA, used instead of unsupervised LSA, showed improvement on the results but considering the added complexity the benefit is debatable. ASA performance highly depends on which resources and MSRs are used; and experimenting with other resources and MSRs is a part of the future work. The source code and outputs of each experiment are available at: https://code.google.com/p/rainbow-nlp/.

# 6 CONCLUSION AND FUTURE WORK

In this study a new hybrid method for semantic relatedness calculation between textual concepts was introduced. The proposed method helps combine various heterogeneous resources and methods to achieve the best performance. The performance however depends on what the included methods and resources are in the hybrid model. The previous chapters show that the hybrid model outperforms each individual method used in the model in both extrinsic and extrinsic evaluations. ASA was evaluated in two applications: named entity normalization and gene function extraction. Furthermore, semantic relatedness application is mainly in information retrieval and relevance, and the two applications demonstrated the power of semantic relatedness in proposing solution for various problems.

## ASA Conclusion and Future Work

In this case, we introduced ASA, a hybrid framework, to combine existing MSR methods and resources in order to achieve semantic similarity scores with the highest correlation to human assigned scores. The experimental results over the UMNSRS benchmark show that the ASA outperforms all other evaluated methods in respect to correlation. It yields the highest completeness by combining all evaluated resources and techniques. We also showed that the term vectors generated by the Skip-gram model, did not perform as well as the traditional LSA technique using the same corpus.

Our future studies will include experiments to test ASA performance on other benchmarks and incorporate other categories of MSR methods such as graph-based methods. In addition, we are keen to investigate adding some characteristics of the corpora as features in the regression model to prevent it from undervaluing smaller corpora for the rare cases

in which corpus would be useful. Although, the proposed method is evaluated only for the biomedical concepts, the proposed approach is domain-independent. Performing experiments in other domains is part of our future work. Even though ASA showed improvement over each evaluated individual MSR, the need to have training set might make it less functional in cases where no annotation is available. As consequence, one of the important future studies is to train ASA and use it in a different context; and see how the trained model can be generalized. Another interesting question to answer is to find the minimum required size of the training set. The implementation of the proposed method is open source and publicly available at: https://github.com/ehsane/octopus-semantic-similarity

**ADR Normalization Conclusion and Future Work**

In this case, we proposed a natural language processing pipeline for the problem of normalizing extracted mentions of ADRs from colloquial texts to UMLS concepts. We compared semantic similarity techniques and evaluated a hybrid approach. The hybrid approach shows improvement over a single similarity technique (LSA). Tweets, like other informal texts, require heavy pre-processing and cleaning. The errors of the system could be reduced by applying more advanced pre-processing like spell correction. This is the first effort for ADR normalization and can provide a baseline for future works. Different social media promote different language patterns and evaluating the proposed pipeline on other social media corpora is part of our future work, which can show how much the sub-language used in different social media services are similar.

### Gene Function Extraction Conclusion and Future Work

This case presents an unsupervised approach based on LSA and a supervised approach using ASA for gene function extraction from biomedical literature. The goals of the comparison were to replace an MSR method in an existing solution with ASA and evaluate how much the performance of the overall system will change. By replacing LSA with ASA, superior results were achieved. The improvement was more clear when exact match evaluation was used. At the same time ASA added more process complexity since it was a supervised approach and requires training. Considering that ASA is a hybrid model, its performance can change by using different resources and MSRs. Investigating the effect of adding more resources or MSRs on the performance is an interesting future topic. In addition it is interesting to see how much the size of resources used in ASA can change the result.

## Conclusion

This study focused on finding semantic relatedness scores of two textual entities and compared multiple existing methods for this goal. As shown in the past, hybrid models yielded better results than each individual method but combining different methods is not easy tasks since each semantic relatedness method need specific type of resource. Combining this heterogeneous resources is either impossible or requires a lot of manual work. This thesis proposes a new hybrid model which can be automatically combined with various heterogeneous semantic relatedness methods and resources. This yields benefits from all types of resources and methods through calculating a single relatedness score. Also the proposed hybrid model is flexible in learning the best combination of methods and resources in a specific problem area.

To show the effectiveness of the proposed hybrid model, it was evaluated in three different setting:

1. Correlation analysis with expert relatedness rating for medical phrases (intrinsic evaluation);

2. Adverse Drug Reaction normalization task;

3. Gene function extraction from biomedical literature.

In all three case studies the proposed hybrid model outperformed individual methods. Results indicate the effectiveness of the hybrid model over individual method, which confirms the expected superiority of hybrid models. This work novelty is to introduce a dynamic hybrid model which can benefit from any new methods or resources with minimum effort. With this in mind, ASA may boost any applications of semantic relatedness by providing a framework to find optimal combination of available methods and resources for a specific purpose. ASA source code is publicly available (https://github.com/ehsane/octopus-semantic-similarity) which provides a framework for the communities to explore other methods and resources and to solve various problems in semantic relatedness and biomedical science.

## REFERENCES


[1] Pesquita C, Faria D, Falcão AO, Lord P, Couto FM. Semantic similarity in biomedical ontologies. PLoS Comput Biol 2009;5:e1000443. doi:10.1371/journal.pcbi.1000443.

[2] Harabagiu S, Bejan CA, Mor P. Shallow Semantics for Relation Extraction. Lang Learn n.d.:1–6.

[3] Jonnalagadda S, Leaman R, Cohen T, Gonzalez G. A Distributional Semantics Approach to Simultaneous Recognition of Multiple Classes of Named Entities. Comput Linguist Intell Text Process 2010.

[4] Hliaoutakis A, Varelas G, Voutsakis E, Petrakis EGM, Milios E. Information Retrieval by Semantic Similarity. Int J Semant Web Inf Syst 2006;2:55–73. doi:10.4018/jswis.2006070104.

[5] Alexander Budanitsky AB. Lexical Semantic Relatedness and Its Application in Natural Language Processing n.d.

[6] Cao Y, Liu F, Simpson P, Antieau L, Bennett A, Cimino JJ, et al. AskHERMES: An online question answering system for complex clinical questions. J Biomed Inform 2011;44:277–88. doi:10.1016/j.jbi.2011.01.004.

[7] Cohen AM, Hersh WR. A survey of current work in biomedical text mining. Brief Bioinform 2005;6:57–71.

[8] Ostler E. STEM Education: An Overview of Contemporary Research, Trends, and Perspectives. 2015.

[9] The American Heritage Dictionary of the English Language. Fifth. Houghton Mifflin Harcourt Publishing Company; n.d.

[10] Harris ZS. Distributional structure n.d.

[11] Sahlgren M. The distributional hypothesis. Ital J Linguist 2008:1–18.

[12] Zhou X, Zhang X, Hu X. Dragon Toolkit: Incorporating auto-learned semantic knowledge into large-scale text retrieval and mining. ictai, IEEE Computer Society; 2007, p. 197–201. doi:10.1109/ICTAI.2007.117.

[13] Athenikos SJ, Han H. Biomedical question answering: a survey. Comput Methods Programs Biomed 2010;99:1–24. doi:10.1016/j.cmpb.2009.10.003.

[14] Pivovarov R, Elhadad N. A hybrid knowledge-based and data-driven approach to

identifying semantically similar concepts. J Biomed Inform 2012;45:471–81. doi:10.1016/j.jbi.2012.01.002.

[15] Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R. Indexing by latent semantic analysis. J Am Soc Inf Sci 1990;41:391–407.

[16] Turney PD. Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. Lect Notes Comput Sci Vol 2167 2001.

[17] Lemaire B, Denhière G. Incremental construction of an associative network from a corpus. Proc 26th Annu Meet Cogn Sci Soc 2004:825–30.

[18] Budiu R, Royer C, Pirolli P. Modeling Information Scent: A Comparison of LSA, PMI and GLSA Similarity Measures on Common Tests and Corpora. Proc RIAO'07 Pittsburgh, PA 2007.

[19] Matveeva I, Farahat A, Royer C. Term representation with generalized latent semantic analysis. RANLP (Recent Adv Nat Lang Process 2005.

[20] Zeno S, Ivens S, Millard R, Duvvuri R. The educator's word frequency guide. Touchstone Applied Science Associates (TASA) 1995.

[21] Cho J, Garcia-Molina H, Haveliwala T, Lam W, Paepcke A, Raghavan S, et al. Stanford WebBase components and applications. ACM Trans Internet Technol 2006;6:153–86. doi:10.1145/1149121.1149124.

[22] Emadzadeh E, Nikfarjam A, Muthaiyah S. A comparative study on Measure of Semantic Relatedness function. 2010 2nd Int. Conf. Comput. Autom. Eng., IEEE; 2010, p. 94–7. doi:10.1109/ICCAE.2010.5451991.

[23] Muthaiyah S, Kerschberg L. A Hybrid Ontology Mediation Approach for the Semantic Web. Int J E-Bus Res 2008;4:79–91.

[24] Pedersen T, Pakhomov SVS, Patwardhan S, Chute CG. Measures of semantic similarity and relatedness in the biomedical domain. J Biomed Inform 2007;40:288–99. doi:10.1016/j.jbi.2006.06.004.

[25] Liu Y, McInnes B. Semantic relatedness study using second order co-occurrence vectors computed from biomedical corpora, UMLS and WordNet. Proc 2nd … 2012:363. doi:10.1145/2110363.2110405.

[26] Petrakis EGM, Varelas G, Hliaoutakis A, Raftopoulou P. X-Similarity: Computing Semantic Similarity between concepts from different ontologies. J Digit Inf Manag 2006;4:233–7.

[27] Rodríguez M, Egenhofer M. Determining semantic similarity among entity classes from different ontologies. IEEE Trans Knowl Data Eng 2003;15.

[28] Bandar Z a., McLean D. An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans Knowl Data Eng 2003;15:871–82. doi:10.1109/TKDE.2003.1209005.

[29] Stetson PD, Johnson SB, Scotch M, Hripcsak G. The sublanguage of cross-coverage. Proc AMIA Symp 2002:742–6. doi:D020002450 [pii].

[30] Jonnalagadda S, Cohen T, Wu S, Gonzalez G. Enhancing clinical concept extraction with distributional semantics. J Biomed Inform 2011. doi:10.1016/j.jbi.2011.10.007.

[31] Yi E, Lee GG, Song Y, Park S. SVM-Based Biological Named Entity Recognition Using Minimum Edit-Distance Feature Boosted by Virtual Examples 2005:807–14.

[32] Robert Leaman GG. BANNER: An executable survey of advances in biomedical named entity recognition n.d.

[33] Ginter F, Heimonen J, Pyysalo S, Salakoski T. Learning to Extract Biological Event and Relation Graphs. Computer (Long Beach Calif) 2009:18–25.

[34] Mcclosky D, Surdeanu M, Manning CD. Event Extraction as Dependency Parsing for BioNLP 2011. Gene Expr 2011.

[35] Kim JD, Wang Y, Takagi T, Yonezawa A. Overview of genia event task in bionlp shared task 2011. ACL HLT 2011 2011:7.

[36] Melli G, Shi Z, Wang Y, Liu Y, Sarkar A, Popowich F, et al. Description of S QUASH , the SFU Question Answering Summary Handler for the DUC-2006 Summarization Task 2006.

[37] Yamamoto Y, Takagi T. A sentence classification system for multi biomedical literature summarization. Data Eng Work 2005 … 2005.

[38] Lu Z, Hirschman L. Biocuration workflows and text mining: overview of the BioCreative 2012 Workshop Track II. Database (Oxford) 2012;2012:bas043. doi:10.1093/database/bas043.

[39] Deshazo JP, Lavallie DL, Wolf FM. Publication trends in the medical informatics literature: 20 years of "Medical Informatics" in MeSH. BMC Med Inform Decis Mak 2009;9:7. doi:10.1186/1472-6947-9-7.

[40] Lu Z. PubMed and beyond: a survey of web tools for searching biomedical literature. Database (Oxford) 2011;2011:baq036. doi:10.1093/database/baq036.

[41] Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet 2000;25:25–9. doi:10.1038/75556.

[42] Mao Y, Auken K Van. The gene ontology task at biocreative IV. Proc BioCreative IV Work Bethesda, USA 2013;1:119–27.

[43] Auken, K.V., Schaeffer, M.L., McQuilton P. Corpus Construction for the BioCreative IV GO Task. Proc BioCreative IV Work Bethesda, USA 2013.

[44] Blaschke C, Leon EA, Krallinger M, Valencia A. Evaluation of BioCreAtIvE assessment of task 2. BMC Bioinformatics 2005;6 Suppl 1:S16. doi:10.1186/1471-2105-6-S1-S16.

[45] Chiang J, Yu H. Extracting functional annotations of proteins based on hybrid text mining approaches. Proc BioCreAtIvE Chall Eval Work 2004.

[46] Couto FM, Silva MJ, Coutinho PM. Finding genomic ontology terms in text using evidence content. BMC Bioinformatics 2005;6 Suppl 1:S21. doi:10.1186/1471-2105-6-S1-S21.

[47] Ray S, Craven M. Learning statistical models for annotating proteins with function information using biomedical text. BMC Bioinformatics 2005;6 Suppl 1:S18. doi:10.1186/1471-2105-6-S1-S18.

[48] Kaplan AM, Haenlein M. Users of the world, unite! The challenges and opportunities of Social Media. Bus Horiz 2010;53:59–68. doi:10.1016/j.bushor.2009.09.003.

[49] Luppicini R. Handbook of Research on Technoself: Identity in a Technological Society n.d.

[50] Jacob F. Measuring the degree of corporate social media use. Int J Mark Res 2015;57:257–75.

[51] Berners-Lee TJ. The world-wide web. Comput Networks ISDN Syst 1992;25:454–9. doi:10.1016/0169-7552(92)90039-S.

[52] Struik LL, Baskerville NB. The Role of Facebook in Crush the Crave, a Mobile- and Social Media-Based Smoking Cessation Intervention: Qualitative Framework Analysis of Posts. J Med Internet Res 2014;16:e170. doi:10.2196/jmir.3189.

[53] Hanson CL, Cannon B, Burton S, Giraud-Carrier C. An exploration of social circles and prescription drug abuse through Twitter. J Med Internet Res 2013;15:e189. doi:10.2196/jmir.2741.

[54] Nakhasi A, Passarella RJ, Bell SG, Paul MJ, Dredze M, Pronovost PJ. Malpractice and Malcontent : Analyzing Medical Complaints in Twitter. AAAI Tech report,Information Retr Knowl Discov Biomed Text, 2010:1–2.

[55] Nikfarjam A, Gonzalez GH. Pattern mining for extraction of mentions of Adverse Drug Reactions from user comments. AMIA Annu Symp Proc 2011;2011:1019–26.

[56] Benton A, A, C, B, Leonard CE. Identifying potential adverse effects using the web: A new approach to medical hypothesis generation. J Biomed Inform 2011.

[57] Yates A, Goharian N. ADRTrace: detecting expected and unexpected adverse drug reactions from user reviews on social media sites. Adv Inf Retr 2013:816–9.

[58] Ginn R, Pimpalkhute P, Nikfarjam A, Patki A, Oconnor K, Sarker A, et al. Mining Twitter for Adverse Drug Reaction Mentions : A Corpus and Classification Benchmark. Proc. Fourth Work. Build. Eval. Resour. Heal. Biomed. Text Process., 2014.

[59] Li N, Wu DD. Using text mining and sentiment analysis for online forums hotspot detection and forecast. Decis Support Syst 2010;48:354–68. doi:10.1016/j.dss.2009.09.003.

[60] Lin F-R, Hsieh L-S, Chuang F-T. Discovering genres of online discussion threads via text mining. Comput Educ 2009;52:481–95. doi:10.1016/j.compedu.2008.10.005.

[61] Yuce S, Agarwal N, Wigand RT. Social Computing, Behavioral-Cultural Modeling and Prediction. vol. 7812. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. doi:10.1007/978-3-642-37210-0.

[62] Lindquist M. The need for definitions in pharmacovigilance. Drug Saf 2007;30:825–30. doi:10.2165/00002018-200730100-00001.

[63] Harpaz R, DuMouchel W, Shah NH, Madigan D, Ryan P, Friedman C. Novel data-mining methodologies for adverse drug event discovery and analysis. Clin Pharmacol Ther 2012;91:1010–21. doi:10.1038/clpt.2012.50.

[64] Sarker A, Nikfarjam A, O'Connor K, Ginn R, Gonzalez G, Upadhaya T, et al. Utilizing social media data for pharmacovigilance: A review. J Biomed Inform 2015;54:202–12. doi:10.1016/j.jbi.2015.02.004.

[65] Rada R, Mili H, Bicknell E, Blettner M. Development and application of a metric on semantic nets. IEEE Trans Syst Man Cybern 1989;19:17–30. doi:10.1109/21.24528.

[66] Leacock C, Chodorow M. Combining local context with WordNet similarity for word sense identification 1998.

[67] Wu Z, Palmer M. Verbs semantics and lexical selection. Proc. 32nd Annu. Meet. Assoc. Comput. Linguist. -, Morristown, NJ, USA: Association for Computational Linguistics; 1994, p. 133–8. doi:10.3115/981732.981751.

[68] Patwardhan S, Pedersen T. Using WordNet-based Context Vectors to Estimate the Semantic Relatedness of Concepts. 11th Conf. Eur. Chapter Assoc. Comput. Linguist., vol. 1501, 2006, p. 1–8. doi:citeulike-article-id:1574418.

[69] Lesk M. Automatic sense disambiguation using machine readable dictionaries. Proc. 5th Annu. Int. Conf. Syst. Doc. - SIGDOC '86, New York, New York, USA: ACM Press; 1986, p. 24–6. doi:10.1145/318723.318728.

[70] Banerjee S, Pedersen T. An adapted Lesk algorithm for word sense disambiguation using WordNet. Comput Linguist Intell Text … 2002:136–45. doi:10.1007/3-540-45715-1_11.

[71] Cilibrasi RL, Vitanyi PMB. The Google Similarity Distance. IEEE Trans Knowl Data Eng 2007;19.

[72] Harispe S, Ranwez S, Janaqi S, Montmain J. The semantic measures library and toolkit: fast computation of semantic similarity and relatedness using biomedical ontologies. Bioinformatics 2013;30.

[73] Joachims T. Making large scale SVM learning practical. Adv. kernel methods, MIT Press Cambridge, MA, USA; 1999, p. 169–84.

[74] Landauer TK and STD. A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. Psychol Rev 1997.

[75] Wikipedia. Singular Value Decomposition n.d. http://en.wikipedia.org/wiki/Singular_value_decomposition.

[76] Jiang JJ, Conrath DW. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. Int. Conf. Res. Comput. Linguist. (ROCLING X), 1997, p. 9008+.

[77] Lin D. An Information-Theoretic Definition of Similarity. Proc Fifteenth Int Conf Mach Learn 1998.

[78] Resnik P. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. Proc. 14th Int. Jt. Conf. Artif. Intell., 1995, p. 448–53.

[79] Sarker A, Gonzalez G. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. J Biomed Inform 2015;53:196–207. doi:10.1016/j.jbi.2014.11.002.

[80] Nikfarjam A, Sarker A, O'Connor K, Ginn R, Gonzalez G. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. J Am Med Informatics Assoc 2015;22:671–81. doi:10.1093/jamia/ocu041.

[81] Nelson, Stuart J.; Powell, Tammy; Humphreys BL. The Unified Medical Language System (UMLS) Project. 2002.

[82] Jimeno A, Jimenez-Ruiz E, Lee V, Gaudan S, Berlanga R, Rebholz-Schuhmann D. Assessment of disease named entity recognition on a corpus of annotated sentences. BMC Bioinformatics 2008;9 Suppl 3:S3. doi:10.1186/1471-2105-9-S3-S3.

[83] Leaman R, Miller C. Enabling Recognition of Diseases in Biomedical Text with Machine Learning : Corpus and Benchmark. Proc. 3rd Int. Symp. Lang. Biol. Med., 2009, p. 82–9.

[84] Kang N, Singh B, Afzal Z, van Mulligen EM, Kors JA. Using rule-based natural language processing to improve disease normalization in biomedical text. J Am Med Inform Assoc 2013;20:876–81. doi:10.1136/amiajnl-2012-001173.

[85] Hirschman L, Yeh A, Blaschke C, Valencia A. Overview of BioCreAtIvE: critical assessment of information extraction for biology. BMC Bioinformatics 2005;6:S1. doi:10.1186/1471-2105-6-S1-S1.

[86] Kim J-D, Ohta T, Pyysalo S, Kano Y, Tsujii J. Overview of BioNLP'09 shared task on event extraction. Proc. Work. BioNLP Shar. Task - BioNLP '09, Morristown, NJ, USA: Association for Computational Linguistics; 2009, p. 1. doi:10.3115/1572340.1572342.

[87] Clarke CLA, Craswell N, Voorhees EM. Overview of the TREC 2012 Web Track. TREC, 2012, p. 1–8.

[88] Uzuner O, South BR, Shen S, Duvall SL. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. J Am Med Inform Assoc 2011:552–7. doi:10.1136/amiajnl-2011-000203.

[89] Leaman R, Islamaj Dogan R, Lu Z. DNorm: disease name normalization with pairwise learning to rank. Bioinformatics 2013;29:2909–17. doi:10.1093/bioinformatics/btt474.

[90] Bashyam V, Divita G, Bennett DB, Browne AC, Taira RK. A normalized lexical lookup approach to identifying UMLS concepts in free text. Stud Health Technol Inform 2007;129:545–9.

[91] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Proc AMIA Symp 2001:17–21. doi:D010001275 [pii].

[92] Huang M, Névéol A, Lu Z. Recommending MeSH terms for annotating biomedical articles. J Am Med Inform Assoc 18:660–7. doi:10.1136/amiajnl-2010-000055.

[93] Huang M, Liu J, Zhu X. GeneTUKit: a software for document-level gene

normalization. Bioinformatics 2011;27:1032–3. doi:10.1093/bioinformatics/btr042.

[94] Sullivan R, Leaman R, Gonzalez G. The DIEGO Lab Graph Based Gene Normalization System. 2011 10th Int. Conf. Mach. Learn. Appl. Work., vol. 2, IEEE; 2011, p. 78–83. doi:10.1109/ICMLA.2011.140.

[95] Buyko E, Tomanek K, Hahn U. Resolution of Coordination Ellipses in Complex Biological Named Entity Mentions Using Conditional Random Fields. ISMB BioLink SIG 2007:163–71.

[96] Tsuruoka Y, McNaught J, Tsujii J, Ananiadou S. Learning string similarity measures for gene/protein name dictionary look-up using logistic regression. Bioinformatics 2007;23:2768–74. doi:10.1093/bioinformatics/btm393.

[97] Wermter J, Tomanek K, Hahn U. High-performance gene name normalization with GENO. Bioinformatics 2009;25:815–21. doi:10.1093/bioinformatics/btp071.

[98] Patwardhan S, Banerjee S, Pedersen T. Using Measures of Semantic Relatedness for Word Sense Disambiguation. Proc. Fourth Int. Conf. Intell. Text Process. Comput. Linguist., vol. 4, Springer-Verlag; 2003, p. 241–57. doi:10.1007/3-540-36456-0_24.

[99] Han B, Cook P, Baldwin T. Lexical normalization for social media text. ACM Trans Intell Syst Technol 2013;4:1–27. doi:10.1145/2414425.2414430.

[100] Choudhury M, Saraf R, Jain V, Mukherjee A, Sarkar S, Basu A. Investigation and modeling of the structure of texting language. Int J Doc Anal Recognit 2007;10:157–74. doi:10.1007/s10032-007-0054-0.

[101] Cook P, Stevenson S. An Unsupervised Model for Text Message Normalization. Proc. Work. Comput. Approaches to Linguist. Creat., Association for Computational Linguistics; 2009, p. 71–8. doi:10.3115/1642011.1642021.

[102] Xue Z, Yin D, Davison B. Normalizing Microtext. Anal Microtext 2011:74–9.

[103] Liu F, Weng F, Jiang X. A Broad-Coverage Normalization System for Social Media Language. Proc 50th Annu Meet Assoc Comput Linguist Vol 1 Long Pap 2012:1035–44.

[104] Sellberg L, Jönsson A. Using Random Indexing to improve Singular Value Decomposition for Latent Semantic Analysis. LREC 2008:2335–8.

[105] Wild F, Stahl C, Stermsek G, Neumann G. Parameters Driving Effec- Tiveness of Automated Essay Scoring. Inf Syst J 2005;80:485–94.

[106] Hofmann T. Probabilistic latent semantic analysis. Proc Uncertain Artif Intell 1999.

[107] Emadzadeh E, Nikfarjam A, Ginn RE, Gonzalez G. Unsupervised gene function extraction using semantic vectors. Database 2014;2014:bau084–bau084. doi:10.1093/database/bau084.

[108] Mao Y, Van Auken K, Li D, Arighi CN, McQuilton P, Hayman GT, et al. Overview of the gene ontology task at BioCreative IV. Database (Oxford) 2014;2014. doi:10.1093/database/bau086.

[109] Comeau DC, Islamaj Doğan R, Ciccarese P, Cohen KB, Krallinger M, Leitner F, et al. BioC: a minimalist approach to interoperability for biomedical text processing. Database (Oxford) 2013;2013:bat064. doi:10.1093/database/bat064.

[110] Widdows D, Cohen T. The Semantic Vectors Package: New Algorithms and Public Tools for Distributional Semantics. IEEE Fourth Int Conf Semant Comput 2010:9–15. doi:10.1109/ICSC.2010.94.

[111] Kanerva P, Kristofersson J, Holst A. Random indexing of text samples for latent semantic analysis. Proc 22nd Annu Conf Cogn Sci Soc 2000.

[112] Porter MF. An algorithm for suffix stripping. Progr Electron Libr Inf Syst 1993;14:130–7.

[113] Pakhomov SVS, Pedersen T, McInnes B, Melton GB, Ruggieri A, Chute CG. Towards a framework for developing semantic relatedness reference standards. J Biomed Inform 2011;44:251–65. doi:10.1016/j.jbi.2010.10.004.

[114] Garla VN, Brandt C. Semantic similarity in the biomedical domain: an evaluation across knowledge sources. BMC Bioinformatics 2012;13:261. doi:10.1186/1471-2105-13-261.

[115] Rubenstein H, Goodenough JB. Contextual correlates of synonymy. Commun ACM 1965;8.

[116] Miller G, Charles W. Contextual correlates of semantic similarity. Lang Cogn Process 1991;6:1–28. doi:10.1080/01690969108406936.